NPS ARCHIVE
1967
SEILER, R.

THE EFFECTS OF POLYNOMIAL APPROXIMATION IN FEEDFORWARD COMPENSATION OF CONTROL SYSTEMS WITH TIME DELAY

RICHARD WAYNE SEILER

THE EFFECTS OF POLYNOMIAL APPROXIMATION IN FEEDFORWARD COMPENSATION OF CONTROL SYSTEMS WITH TIME DELAY

by

Richard Wayne Seiler
Lieutenant, United States Navy
B.S.E.E., Bradley University, 1962

[redacted]

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1967

S AIRLINE

oT

ILLER, R.

~~54~~

NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIF. 93940

ABSTRACT

In recent years, a large increase in the number of computer controlled systems and processes has dictated the use of devices with inherently long time delays. The result has been unstable control systems. This thesis develops two methods of feedforward compensation which cope with this problem. The Pade' method of polynomial approximation is used to formulate a number of transfer functions which can be used in the two feedforward compensation schemes. To choose the best transfer function for a particular problem, a minimum integral square error analysis technique is developed for use in the frequency domain. Computer simulations are used to verify the effectiveness of the two compensation schemes.

Table of Contents

Chapter Page

Abstract 2

Table of Contents 3

List of Tables 5

List of Illustrations 7

I Introduction 9

II Feedforward Compensation of Systems with Time Delay 15

Methods of Compensation 15

Summary 18

III Design of Optimal Feedforward Filters 19

The Pade' Approximation 19

Nomenclature 20

Derivation of the Desired Filter 21

The Integral Square Error Index (Frequency domain) 25

The Spectral Weighting Function 31

Comparison of Filter Transfer Functions 32

The Integral Square Error Index (Time domain) 39

Other Approximations for $1-e^{-sT}$ 40

Summary 42

IV Experimental Investigations 44

The Plant 44

The Digital Time Delay 44

Simulation of a Typical Pade' Derived Filter 47
The Step Response 49
Sensitivity 55
Summary 57
V Conclusion 58
Topics for Further Study 59
Bibliography 61
Appendix 1 62
Appendix 2 74
Appendix 3 94
Initial Distribution List 125

List of Tables

Table Page

1 Integral Square Errors (Frequency Domain) for Pade' Derived Filter Transfer Functions 34

2 Integral Square Errors (Time Domain) for Several Pade' Transfer Functions 41

3 Integral Square Errors for B_1 to B_5 and C_1 to C_5 41

List of Illustrations

Figure	Page

1 Plant with Time Delay 10

2 Bode Diagram for Eq. (1) 11

3 Buckley System Diagram 13

4 Methods of Feedforward Compensation 16

5 Equivalent Systems 16

6(a) Poles of G_{mn} 22

6(b) Zeros of G_{mn} 23

7 Hypothetical Experiment 28

8 Mean Square Error vs n for Constant m 35

9(a) Power Density Spectrum - G_{m2} 36

9(b) Power Density Spectrum - G_{7n} 37

10 Plant Simulation 45

11 Analog Plant Simulation 46

12 Digital Time Delay 48

13 Analog Simulation of G_{43} 50

14 Step Response - No Compensation 51

15 Step Response - With G_{21} Compensator 52

With G_{43} Compensator 53

With G_{65} Compensator 54

16 Step Response for G_{21} with 2 Second Plant Delay and Variable Filter Delay 56

I. INTRODUCTION

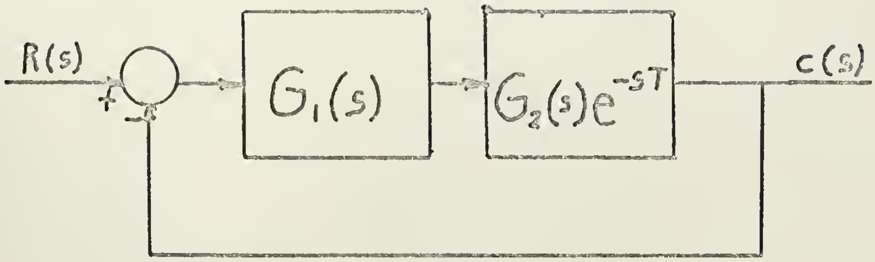
In any control system, with lumped linear elements, the output is theoretically characterized by a response which starts immediately after the application of a transient input. It is well known that this is not really true in many cases, since a number of devices may have a transportation lag or other delay associated with them. If the delay is small, it can usually be ignored. The system is then described by a set of linear differential equations and the classical methods of analysis can be used to study the behavior of the system. In recent years, the large increase in the number of computer controlled systems and processes has dictated the use of devices with inherently long time delays. Examples of such devices would be a digital computer controlled system, where the computation time could be several seconds, or an automated superfractionator, which could have a time delay as long as several hours. A typical system, with unity feedback, is shown in Fig. 1.

When the effects of time delay cannot be ignored, the primary problem is one of system instability. This is easily seen by looking at the Bode diagram shown in Fig. 2. This diagram represents the open loop transfer function for the system in Fig. 1 with

$$G_1(s) = \frac{1}{s(s+1)} \tag{1}$$

and

$$G_2(s) = 1 \tag{2}$$



$$\frac{c(s)}{R(s)} = \frac{G_1(s)G_2(s)e^{-sT}}{1 + G_1(s)G_2(s)e^{-sT}}$$

FIGURE 1.

PLANT WITH TIME DELAY

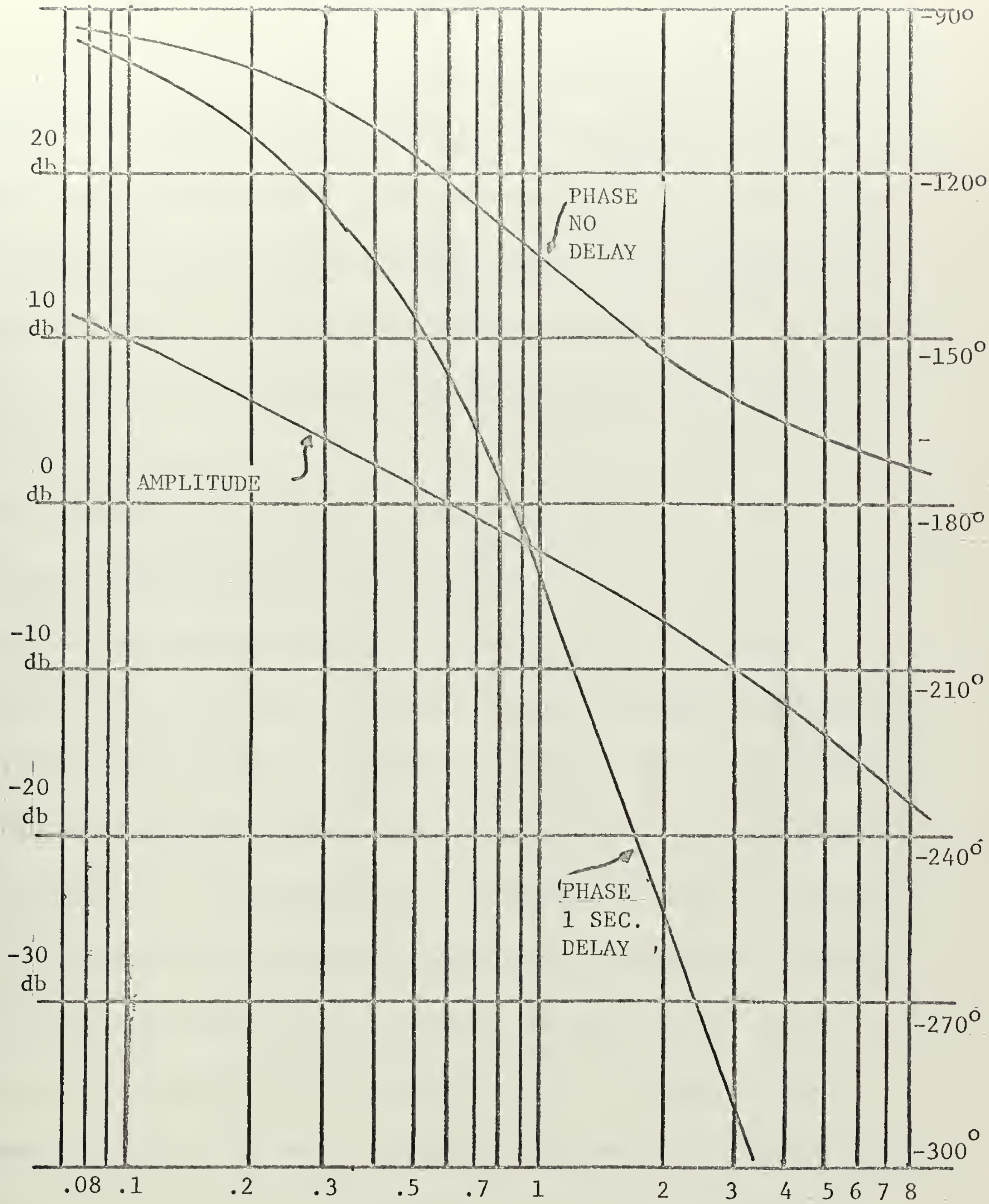


FIGURE 2.

Bode Diagram for Equation 1.

The time delay, T, was set equal to 1 second. The upper phase angle curve represents the system without the time delay. It can be seen that the effect of the delay is to increase the phase lag as the frequency increases. The amount of the increase is equal to the product of the frequency and the time delay. In this particular example, a very stable phase margin of 58.5° has been reduced to 24°. If the time delay is increased to 2 seconds, the system would be unstable. It can be easily concluded that even a modest delay can present formidable stability problems.

This paper presents a method of compensating for time delay in closed loop systems, such as the one shown in Fig. 1. The technique involves feeding a conditioned signal forward, around the element associated with the time delay. The feedforward path may require either active or passive elements to condition the signal. The circuit requirements will be developed. This method is an offspring of one referred to by P. S. Buckley[1,2] as the Smith Linear Predictor. However, Buckley's use of the predictor was not to stabilize a closed loop system. He was concerned with the effects of an external disturbance (noise) being introduced into the loop between the controller and the plant. He assumed the system to be stable before the disturbance was applied. The predictor was only a method for reducing the effect of the disturbance. The Smith Linear Predictor, as presented in Buckley's paper, is shown in Fig. 3.

It can be observed that the circuit requirements for the signal conditioner in the feedforward path cannot be synthesized with lumped,

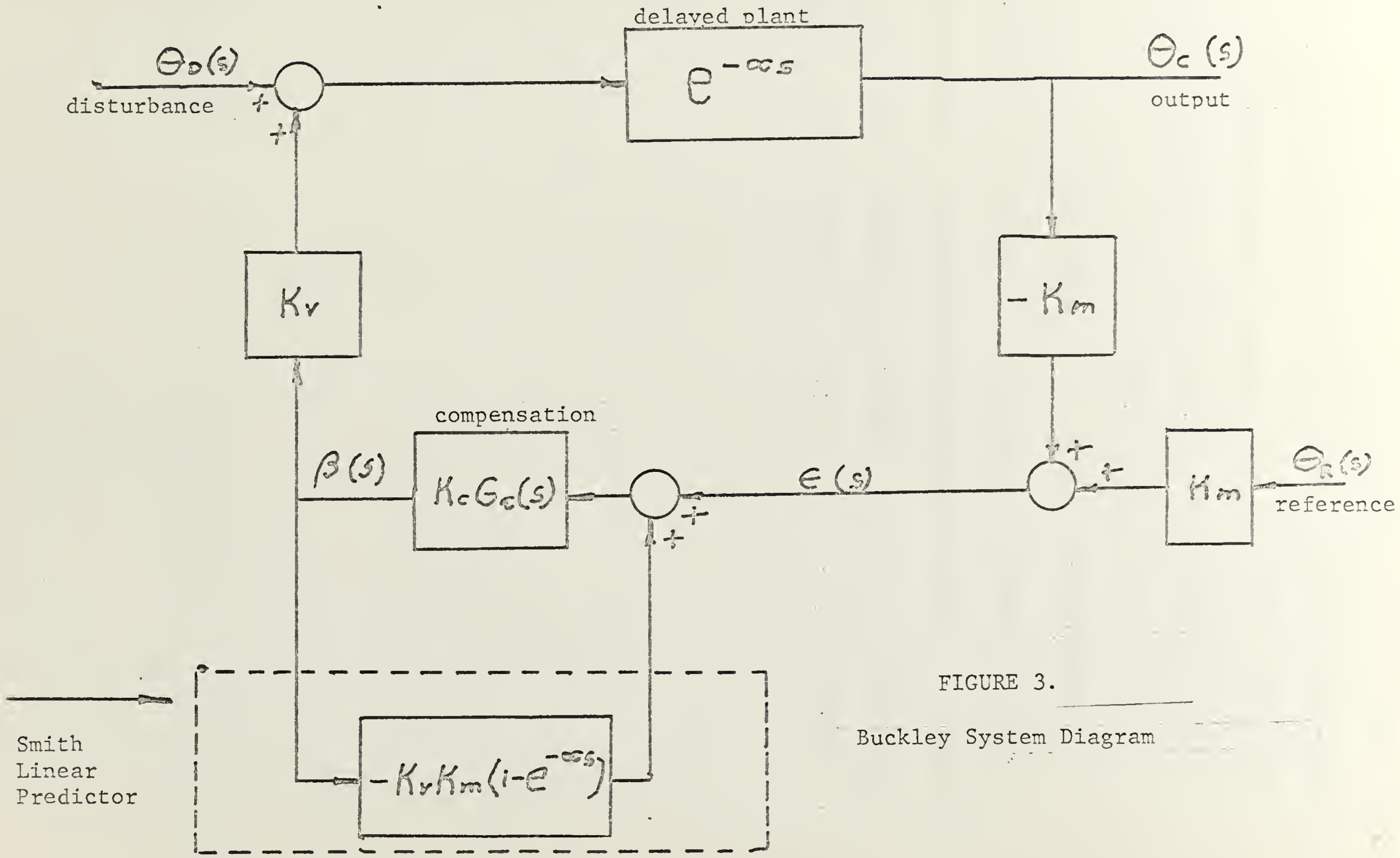


FIGURE 3.

Buckley System Diagram

linear, passive network elements. A suboptimal network will therefore have to be used in the feedforward path. The Pade' method of polynomial approximation will be derived and used to find an acceptable feedforward network. It will be found that a number of acceptable approximations can be formed by the Pade' method to meet the network requirements. The problem will be to choose the best approximation and to find the conditions for which it is the best. In this thesis, a frequency domain technique is devised for choosing the best approximation in a minimum integral square error sense. Experimental examples are then given to show the effectiveness of the approximations and the compensation method.

II. FEEDFORWARD COMPENSATION OF SYSTEMS WITH TIME DELAY

Methods of Compensation. Given a system such as the one shown in Fig. 1, the solution to the stability problem involves the removal of the e^{-sT} term from the denominator of the closed loop transfer function. Two methods of doing this immediately come to mind and are shown in Fig. 4. The transfer function for system 1 (Fig. 4a) can be expressed as

$$\frac{C(s)}{R(s)} = \frac{G_1(s)G_2(s)e^{-sT}}{1 + G_1(s)G_3(s) + G_1(s)G_2(s)e^{-sT}} \tag{3}$$

It can be seen that if $G_3(s)$ is set equal to $G_2(1-e^{-sT})$ the transfer function reduces to

$$\frac{C(s)}{R(s)} = \frac{G_1(s)G_2(s)}{1 + G_1(s)G_2(s)}e^{-sT} \tag{4}$$

It is recognized that this equation represents nothing more than the transfer function of a closed loop system (without a delay element in the loop) cascaded with the time delay.

In system 2 (Fig. 4b), if $G_3(s)$ is again set equal to $G_2(1-e^{-sT})$, the transfer function can be immediately written as

$$\frac{C(s)}{R(s)} = \frac{G_1(s)G_2(s)}{1 + G_1(s)G_2(s)} \tag{5}$$

It can be seen that both techniques meet the objective of removing the e^{-sT} term from the denominator of the closed loop transfer function. Both systems use a standard network, which has a transfer function

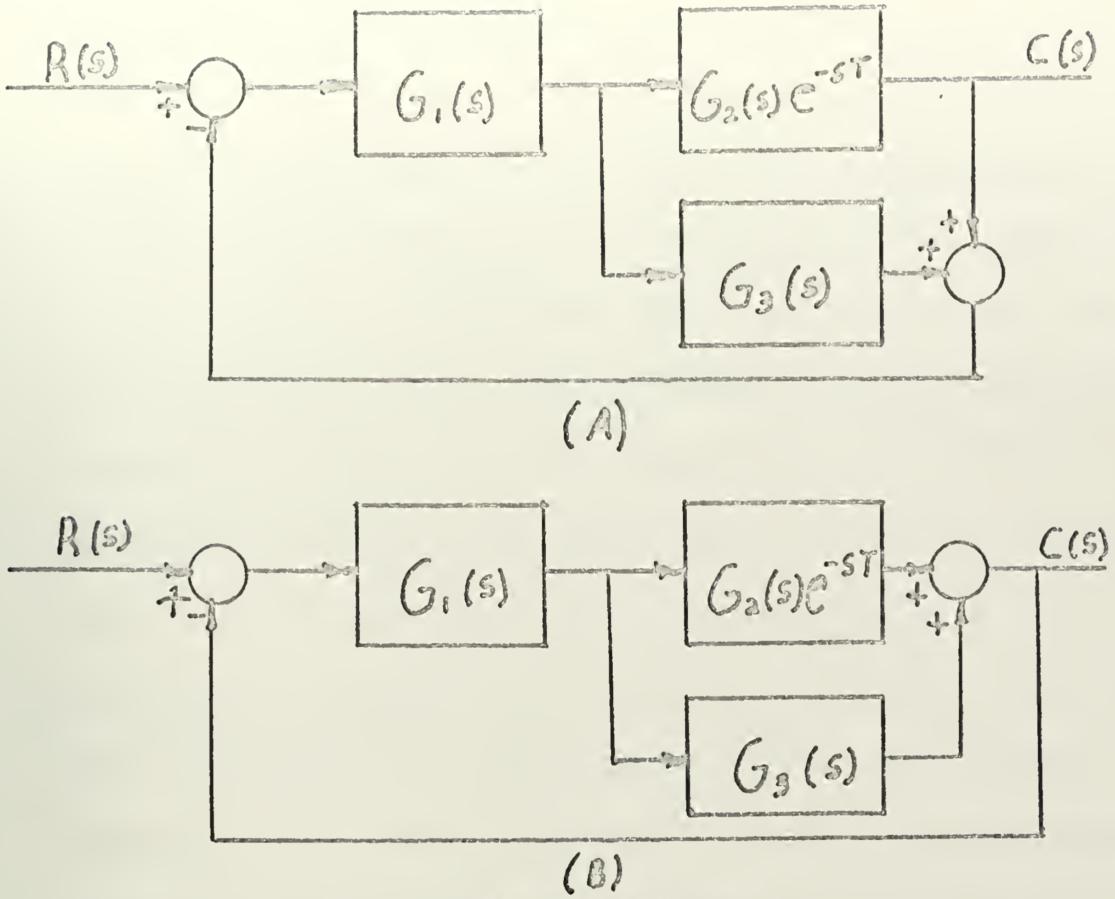


FIGURE 4
METHODS OF FEEDFORWARD COMPENSATION

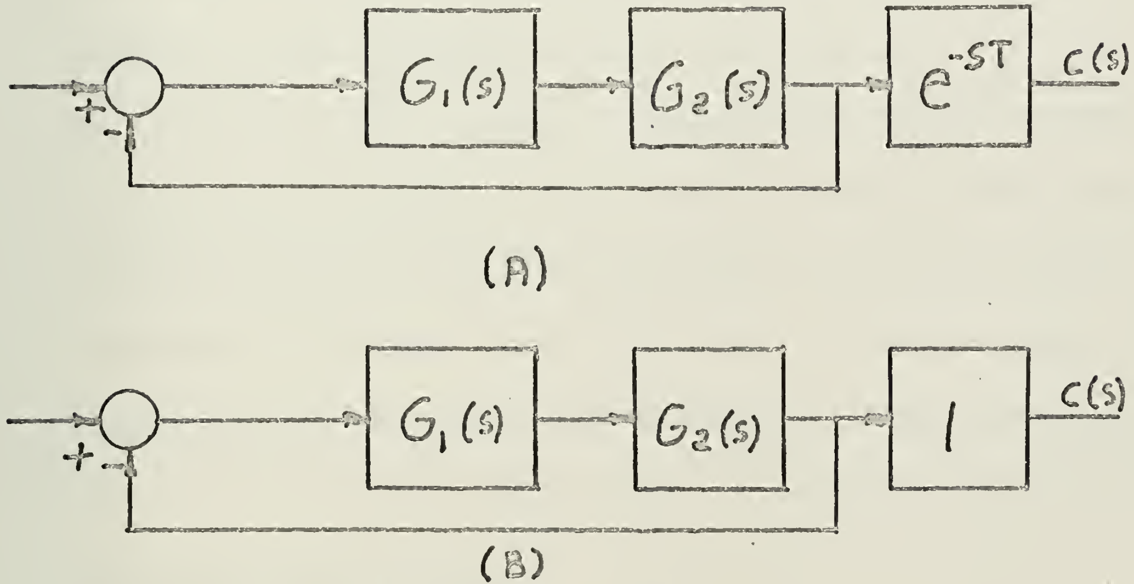


FIGURE 5
EQUIVALENT SYSTEMS

$1-e^{-sT}$, cascaded with a $G_2(s)$ network. This means that the time delay network needs to be synthesized only once, normalized with respect to the time delay, for all problems. The only requirement is an impedance match between the time delay network, $G_2(s)$ network, and any other sections in the feedforward path. There is a subtle difference in the two methods of compensation however. This difference involves the nature of the output. The two systems can be reduced to the equivalent block diagram form shown in Fig. 5. The output of the first system is still delayed by a factor T while the output of the second system starts to respond immediately after the application of the input signal. First impressions would indicate that the second system is better than the first since the time delay has been completely eliminated. However, it remains to be seen from experimental data that this is the case. It is possible that noise attenuation characteristics, speed of response, or other desirable parameter might require use of the first system. A practical consideration might also dictate the use of the first system. If, for example, the time delayed plant is a chemical process, it would be impossible to "mix" the chemical output with an electrical feedforward signal. On the other hand, the electrical feedforward signal could be added to the electrical feedback signal coming from some output sensor, as required in the first system.

The Approximation Requirement. The first step in implementing either of these two systems is to synthesize a network which generates the transfer function $1-e^{-sT}$, which is identical to the transfer function required for the Smith Linear Predictor. Since it is a transcendental

function, some form of approximation must be used if the transfer function is to be realized by lumped passive components. Obviously distributed parameter delay lines can be used but if T is large, as can be expected in a process control, these devices are not practical. A set of linear filter transfer functions can be derived using any of the standard approximation methods. The background, procedure, and accuracy behind the Pade' method of polynomial approximation is contained in the next chapter. Approximations based on this method are used in this thesis for the feedforward network required in the two compensation schemes of Fig. 4.

Summary. The results of this chapter can be summarized as follows:

1. A closed loop control system with time delay can be stabilized if the proper signal can be fed forward around the time delay element.
2. The filter required to generate the feedforward signal has the transfer function

$$G(s) = 1-e^{-sT}$$

3. Approximation techniques must be used to synthesize the required transfer function with lumped, linear components.

The Pade' Approximation. A required step in the synthesis of any function, in lumped, linear, passive, time invariant form, is to insure that the function is rational. The Pade' approximation is a method of representing a polynomial or series of polynomials as a rational function.[3,4] Assume that some function, G(s), is to be approximated by a rational fraction of an n-th order polynomial in s, divided by an m-th order polynomial in s. G(s) could, for example, be the Taylor series expansion of the desired transfer function $1-e^{-sT}$. Thus G(s) can be represented, in general rational form, as

$$G(s) = \frac{a_0 + a_1 s + \cdots + a_n s^n}{b_0 + b_1 s + \cdots + b_m s^m} \tag{6}$$

This expression may be expanded by cross multiplication to give

$$(b_0 + b_1 s + \cdots)(G_0 + G_1 s + \cdots) = (a_0 + a_1 s + a_2 s^2 + \cdots) \tag{7}$$

where G_0, G_1, etc. are the coefficients of the Taylor series expansion of G(s) around the origin in the s-plane. By equating equal powers of s, a set of simultaneous algebraic equations are obtained. For a given number of terms in the Taylor series and for a particular number of terms in the rational fraction, there is a finite number of equations which are solved simultaneously to obtain the a's and b's.

From the s^0 terms $\quad A_0 = b_0G_0$

$$
\begin{aligned}
&s^1 && A_1 = b_1G_0 + b_0G_1 \\
&s^2 && A_2 = b_2G_0 + b_1G_1 + b_0G_2 \\
&s^3 && A_3 = b_3G_0 + b_2G_1 + b_1G_2 + b_0G_3 \\
&\vdots && \vdots \\
&s^n && A_n = b_nG_0 + b_{n-1}G_1 + \cdots + b_0G_n \qquad (8)
\end{aligned}
$$

The rule for finding the s^n equation should be obvious. It is noted that it requires m+n+1 terms of the Taylor series to find n+1 (or n) coefficients for the numerator and m (or m+1) coefficients for the denominator of the rational fraction, and that m+n simultaneous equations are used from (8). This still allows one of the coefficients, say b_0, to be picked arbitrarily and the only difference in the approximations will be a constant multiplier for different values of b_0. It is realized that when the rational fraction is divided out, in ascending powers of s, the first m+n+1 terms of the Taylor series are reproduced exactly. Therefore, the "maximum error" incurred in using the Pade' approximation is no more than the remainder of the Taylor series, taken after m+n+1 terms.

Nomenclature. The following nomenclature will be used in the remainder of this thesis to refer to a particular approximation or transfer function. E_{mn} refers to the Pade' approximation, of the function e^{-sT}, characterized by an n-th order polynomial divided by an m-th order polynomial in s. But e^{-sT} is not the transfer function required in the feedforward path of the systems in Fig. 4. Therefore, G_{mn} will refer to the approximation of the desired transfer function, $1-e^{-sT}$, derived from the Pade'

approximation of e^{-sT} of the same order. For example, if

$$E_{21} = \frac{6-2sT}{6+4sT+s^2T^2} \tag{9}$$

then

$$G_{21} = 1-E_{21} = \frac{6sT+s^2T^2}{6+4sT+s^2T^2} \tag{10}$$

Derivation of the Desired Filter. The Taylor series expansion of e^{-x}, about the value x = 0, is the well known power series

$$e^{-x} = 1-x+\frac{1}{2!}x^2-\frac{1}{3!}x^3+\cdots \tag{11}$$

This is the series that is used to derive rational approximations for the function $1-e^{-sT}$, with $sT = x$. Pade' approximations, for the function e^{-x} are tabulated in many sources.[5] Appendix 1(a) gives such a list of approximations. Appendix 1(b) gives the derived filter transfer functions derived from the appropriate approximation, while Appendix 1(c) gives the derived filter transfer functions in factored form, which is not available in the general literature. The polynomials were factored using the POLYRT subroutine on the CDC 1604 digital computer and rounded off to three decimal places. While the factored form is never used directly in any calculations in this paper, it does give a little insight as to the location of the poles and zeros of each of the filter transfer functions. The poles and zeros are plotted in Fig. 6. There appears to be a pattern to the poles and zeros as the order of s in either the numerator or denominator is increased. This pattern is not known but if it were, it could be used to formulate a general rule for finding the

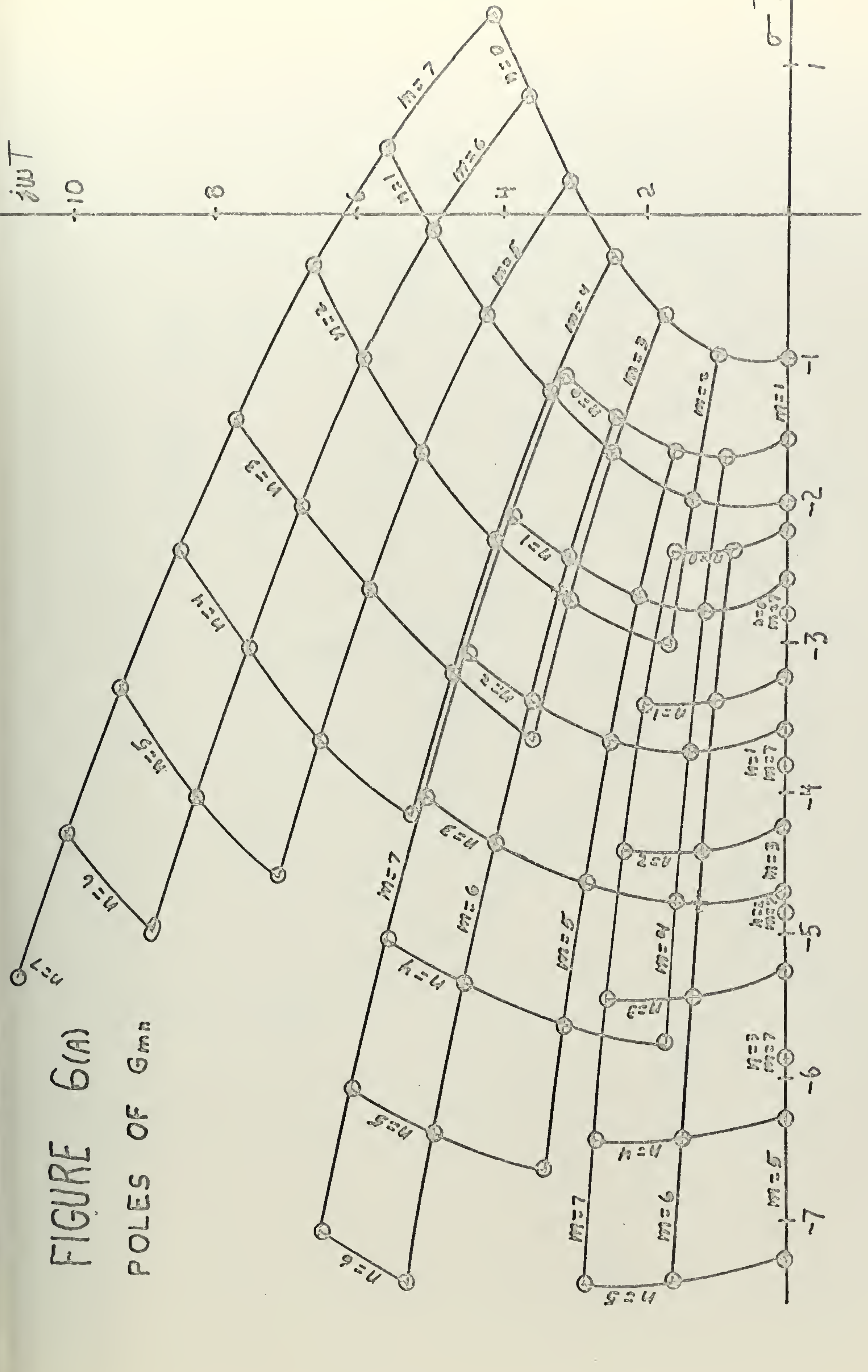


FIGURE 6(A)

POLES OF Gmn

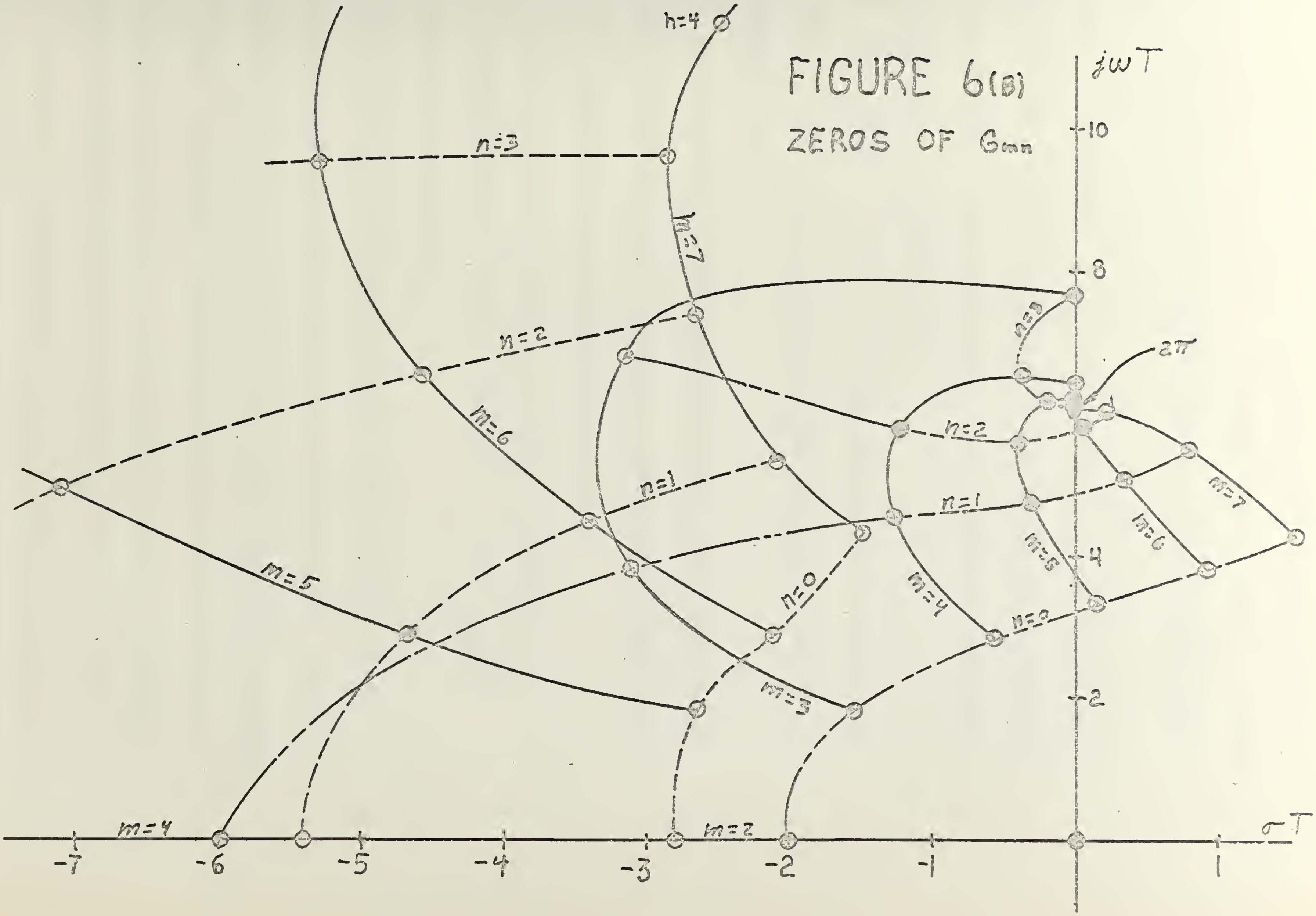


FIGURE 6(B)
ZEROS OF G_{mn}

filter transfer function of any desired order. However, the solution of (8) is quite straight forward in solving for any particular E_{mn} so that no effort was expended to determine an expression for this pole zero pattern.

There is no guarantee that all of the transfer functions found in Appendix 1(b) are realizable with passive elements. The properties of a transfer function, made up of lumped, linear, passive elements are well known but are reiterated here.[6]

1. Poles are in the left half of the s plane or, for the lossless case, on the imaginary axis and simple. Complex poles must be conjugate pairs.
2. Zeros may have any s plane location but complex zeros must occur in conjugate pairs. A minimum phase transfer function has all of its zeros in the left half of the s plane.
3. Poles may not be located at the origin or at infinity.
4. The degree of the numerator may equal but not exceed the degree of the denominator.

Item 4 above rules out all transfer functions where m is less than n. These transfer functions will no longer be considered in this thesis. Referring back to Fig. 6, it is seen that not every Pade' derived transfer function can be synthesized with lumped, linear, passive elements, even though m is greater than n. As the order of s in the denominator is increased, holding the order of s in the numerator constant, the motion of the poles is such that poles begin to appear in the right half of the s plane. This violates rule 1 above and renders these transfer

functions unacceptable as an approximation to the desired filter transfer function, $1-e^{-sT}$. It is also noted that for every transfer function where m equals n, an amplifier is required, for D.C. gain, in addition to the passive elements for synthesis.

There is no indication as to which of the filter transfer functions is "the best." Intuitively, G_{77} should be the best approximation to the ideal transfer function since its derivation required more terms of the power series than any of the others in the list. On the other hand, G_{77} requires more effort to synthesize since it is more complex than the others. It may be that a simple filter will work just as well as a complex one in the feedforward path of the systems in Fig. 4. A measurement of which is the best approximation to the ideal filter transfer function, $1-e^{-sT}$, is attempted below.

The Minimum Integral Square Error Index. The index, chosen to measure which of the filter transfer functions is the best approximation, is the Minimum Integral Square Value of the difference between the responses of the ideal transfer function, $1-e^{-sT}$, and the Pade' derived filter approximation G_{mn}. The integral square error can be calculated in either the time or frequency domains. Although the time domain has received predominant attention in the literature, several advantages uncovered during this investigation urge the use of the frequency domain technique. The first, and most important of the advantages, is the ability to solve for a frequency spectrum which can be later used to find the integral square error for any input. This spectrum, called the weighting function, gives the user a certain feeling for the "quality" of the

network under consideration. When time domain techniques are used, an input signal must immediately be chosen and the result applies for that input only. The result that is obtained is only a number and is not a very satisfying "answer." Without other networks (numbers) for comparison, the results are next to useless. Even with several networks, the user only knows which one has the smallest integral square error, not what causes that error or what effect another input might have. This advantage will be more apparent by comparison of the results of the two techniques. Another advantage of the frequency domain approach (or disadvantage of the time domain method) is not having to take the inverse LaPlace transformation of the network transfer function. Some of the larger Pade' approximations are very difficult and time consuming when it comes to obtaining time domain solutions.

In view of the above arguments, this thesis will primarily use the frequency domain approach to selecting the best Pade' approximation for use as a filter transfer function in the compensation systems of Fig. 4. It is known that the mean square value of a signal is equal to the value of its autocorrelation function at t = 0.[7]

$$\text{mean sq. value of } v(t) = \phi_{vv}(0)$$

$$= \int_{0}^{\infty} v(T)\,v(T+\tau)\,dT \Big|_{\tau=0}$$

$$= \int_{-\infty}^{\infty} \Phi_{vv}(jw)\,dw \qquad (12)$$

where $\Phi_{vv}(jw)$ is referred to as the Power Density Spectrum of the signal. It derives this name from the fact that the integral in (12) equals

the average power dissipated in a 1 ohm resistor by the signal voltage v(t). $\Phi_{vv}(jw)$ can be interpreted as a linear density distribution of the mean square value of the signal along the imaginary axis of the s plane. $\Phi_{vv}(jw)$ is obtained from the relationship

$$\Phi_{vv}(s) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-s\tau} \phi_{vv}(\tau)\, d\tau \tag{13}$$

and then setting s = jw.

Consider the hypothetical experiment shown in Fig. 7. It can be shown[7] that the power density spectrum of the error signal Y(s) can be expressed as

$$\begin{aligned} \Phi_{yy}(s) = \; & H(s)H(-s)\,\Phi_{11}(s) \\ & - H(-s)G(s)\,\Phi_{12}(s) \\ & - H(s)G(-s)\,\Phi_{21}(s) \\ & + G(s)G(-s)\,\Phi_{22}(s) \end{aligned} \tag{14}$$

where

$$H(s) = 1 - e^{-sT} \tag{15}$$

and $\Phi_{11}(s)$ is $1/2\pi$ times the LaPlace transform of the autocorrelation function of $v_1(t)$, $\Phi_{12}(s)$ is $1/2\pi$ times the LaPlace transform of the cross correlation function of $v_1(t)$ and $v_2(t)$, etc. If $v_1(t)$ equals $v_2(t)$, as shown by the dotted lines in Fig. 7, then all of the auto and cross correlation functions in (14) are equal and will be designated by a single $\Phi_{vv}(s)$. It is known that for any arbitrary function F

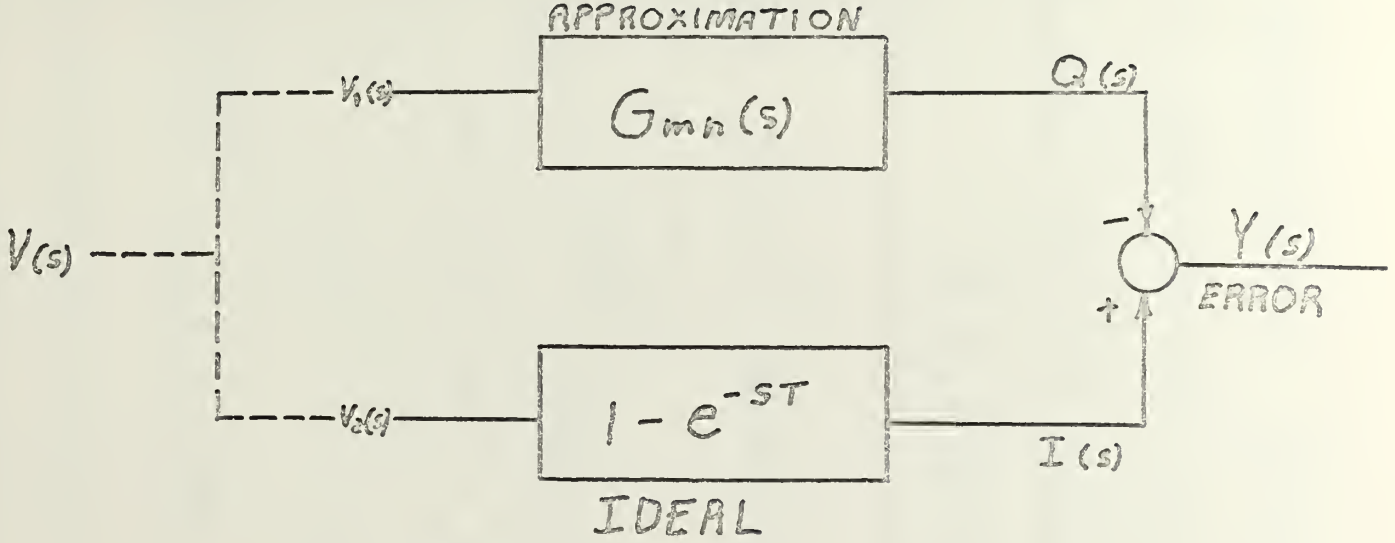


FIGURE 7.

HYPOTHETICAL EXPERIMENT

$$F(jw)F(-jw) = \left|F(jw)\right|^2 \tag{16}$$

Therefore, from (14) with s = jw

$$\Phi_{yy}(jw) = \Phi_{vv}(jw)\Big[\left|H(jw)\right|^2 + \left|G(jw)\right|^2 - 2\,\mathrm{Re}\left\{H(jw)G^*(jw)\right\}\Big] \tag{17}$$

The quantity in brackets in (17) will be referred to as W(jw)

$$W(jw) = H(jw)^2 + G(jw)^2 - 2\,\mathrm{Re}\left[H(jw)G^*(jw)\right] \tag{17a}$$

By trigonometric substitution

$$\begin{aligned} H(jw) &= 1 - e^{-jwT} = 1 - \left[\cos(-wT) + j\,\sin(-wT)\right] \\ &= (1 - \cos wT) + j\,\sin wT \end{aligned} \tag{18}$$

so that

$$\left|H(jw)\right|^2 = 2(1 - \cos wT) \tag{19}$$

From the Pade' approximations, $G_{mn}(s)$ is of the form

$$G(s) = \frac{A_n S^n + A_{n-1} S^{n-1} + \cdots + A_1 S + A_0}{B_m S^m + B_{m-1} S^{m-1} + \cdots + B_1 S + B_0} \tag{20}$$

where m and n depend on the particular approximation being considered. Or in simpler terms

$$G(s) = \frac{M_1(s) + N_1(s)}{M_2(s) + N_2(s)} \tag{21}$$

where the M's refer to the sum of the terms with even powers of s and the N's refer to the sum of the terms with odd powers of s. For real frequencies (when s = jw)

$$G(jw) = \frac{M_1(w) + jN_1(w)}{M_2(w) + jN_2(w)} = A(w) + jB(w) \tag{22}$$

and

$$\left| G(jw) \right|^2 = A^2 + B^2 \tag{23}$$

From (18) and (22), it follows that

$$Re\left[H(jw)\,G^*(jw)\right] = A(w)\left[1 - \cos wT\right] + B(w)\left[\sin wT\right] \tag{24}$$

Substituting (19), (23) and (24) into the bracketed portion of (17) yields

$$W(jw) = 2(1 - \cos wT) + A^2(w) + B^2(w) - 2\left[A(w)(1 - \cos wT) + B(w)\sin wT\right] \tag{25}$$

In shortened form, (17) becomes

$$\Phi_{yy}(jw) = W(jw)\,\Phi_{vv}(jw) \tag{26}$$

W(jw) is given the name Spectral Weighting Function. Referring to Fig. 7, the spectral weighting function does just what its name implies. It weights the magnitude of the power density spectrum of some input signal v(t) to yield the power density spectrum of the error. All that is now required, to determine which of the Pade' derived filter transfer functions is the best, is the substitution of the various approximations, G_{mn}, into (25) to determine W(jw). Multiplying the spectral weighting

function by the power density spectrum of an appropriate input signal yields the power density spectrum of the error for that particular transfer and input function. Integrating the power density spectrum according to (12) yields the integral square value of the error function. The Pade' derived filter transfer function that has the smallest area under the spectrum, i.e. the minimum integral square error, is the one which is the best approximation to the ideal transfer function, $1-e^{-sT}$.

The Spectral Weighting Function. In order to generalize the selection process for any input signal, the spectral weighting functions, for each of the acceptable Pade' derived transfer functions, should be presented in graphical form. This permits the user to know in advance which frequencies of the input signal the various transfer functions weight the most. The transfer function, that best suppresses the frequencies in the input signal, is the one that will have the minimum integral square error. It is significant to note that the best approximation depends upon the spectral distribution of the input signal as well as the spectral weighting function. Due to the large number of Pade' derived filter transfer functions and the complexity of the spectral weighting function equation, these curves were computed using a digital computer. A program was written to compute and plot W(jw) and is presented in Appendix 2(b). The results are shown in Appendix 3. The horizontal axis for the graphs in Appendix 3 is plotted as the $\log_{10}$ of a normalized frequency time delay product, i.e. $x = \log_{10} wT$. In general, the spectral weighting functions for the Pade' derived filter transfer functions have a larger weighting effect at higher frequencies and

appear to act like high pass filters of cutoff frequency, w_0. With the exception of the curves corresponding to the case when m equals n, all of the spectral weighting functions settle to a constant amplitude of one. The spectral weighting function of a transfer function having m equal to n has a constant amplitude oscillation about the value two. If the frequency components of an input signal were lower than some w_0, where the spectral weighing function is zero, then the integral square error would be zero. For such an input signal, the Pade' derived filter transfer functions would be an excellent approximation to the ideal transfer function.

Comparison of Filter Transfer Functions. The input signal that is quite often used to test the performance of a control system or filter is the unit step. The idea is that any signal can be approximated by a sum of little steps. Any filter or system that would perform well with a single step input should work for a sum of steps. A unit step may not, however, be a very good choice for an input signal in the selection process described above. Note that the minimum integral square error is a nonlinear performance index and the principle of superposition cannot be used. The Pade' derived filter transfer function that is selected as the best, based on a unit step, cannot be assumed to be the best for some other input. The minimum integral square error performance index does not, however, rule out the possibility that the best transfer function for a step input may also be best for other inputs. It all depends on the power density spectrum of the input. For the purpose of this paper, a unit step will be used as the input signal when selecting a Pade'

approximation to use as the feedforward filter transfer function. This choice was made simply because the unit step is general and easy to work with. In an actual problem, where the input signal is known, the power density spectrum of the actual input would be used. Also, the unit step makes the examination of experimental results a familiar task. The power density spectrum of a unit step is given by the equation

$$\Phi_{vv}(j\omega) = \frac{1}{2\pi\,\omega^2} \tag{27}$$

The digital computer is also used to multiply the spectral weighting function by the power density spectrum of a unit step and to numerically integrate the resulting area under the power density spectrum of the error. The program that was written to perform this operation is given in appendix 2(c). The integral square error for each of the acceptable Pade' derived transfer functions is tabulated in table 1. The table is broken into several bands to show how the frequencies of the spectrum contribute to the integral square error. Figs. 8 and 9 are graphical representations of the data in table 1. The solid lines in Fig. 8 show the variation in the integral square error as one of the indices (m or n) is varied. Horizontal movement along one of the solid lines from one point to another represents a change in the order of s in the numerator, while holding the order of s in the denominator constant. It is observed that, as long as n is less than m, an increase in the order of s in the numerator results in a decrease in the integral square error. The amount of the decrease becomes smaller as n approaches m. When n becomes equal to m, there is a substantial increase in the integral square error and

Table 1

Integral Square Errors (Frequency Domain) for Pade' Derived Filter Transfer Functions

Step Input -- Time Delay 1 Second

—

Band 0	less than .01 rps	Band 3	10. to 100. rps
Band 1	.01 to 1. rps	Band 4	100. to 1000. rps
Band 2	1. to 10. rps	Band 5	over 1000. rps

—

Band	0	1	2	3	4	5	M.S.E.
G 10	aprx.0	.0165	.1868	.0293	.0029	aprx.0	.2355
G 11	aprx.0	.0003	.2021	.0717	.0057	aprx.0	.2699
G 20	aprx.0	.0016	.1142	.0286	.0029	aprx.0	.1471
G 21	aprx.0	.0000	.0748	.0283	.0029	aprx.0	.1060
G 22	aprx.0	aprx.0	.0974	.0506	.0057	aprx.0	.1537
G 30	aprx.0	aprx.0	.0974	.0286	.0029	aprx.0	.1289
G 31	aprx.0	aprx.0	.0449	.0283	.0029	aprx.0	.0761
G 32	aprx.0	aprx.0	.0369	.0290	.0029	aprx.0	.0687
G 33	aprx.0	aprx.0	.0401	.0605	.0057	aprx.0	.1063
G 40	aprx.0	aprx.0	.1474	.0286	.0029	aprx.0	.1789
G 41	aprx.0	aprx.0	.0347	.0286	.0029	aprx.0	.0662
G 42	aprx.0	aprx.0	.0197	.0296	.0029	aprx.0	.0522
G 43	aprx.0	aprx.0	.0137	.0340	.0029	aprx.0	.0508
G 44	aprx.0	aprx.0	.0095	.0695	.0057	aprx.0	.0809
G 51	aprx.0	aprx.0	.0366	.0288	.0029	aprx.0	.0682
G 52	aprx.0	aprx.0	.0138	.0292	.0029	aprx.0	.0458
G 53	aprx.0	aprx.0	.0064	.0306	.0029	aprx.0	.0398
G 54	aprx.0	aprx.0	.0030	.0344	.0029	aprx.0	.0402
G 55	aprx.0	aprx.0	.0012	.0584	.0057	aprx.0	.0652
G 61	aprx.0	aprx.0	.1366	.0286	.0029	aprx.0	.1681
G 62	aprx.0	aprx.0	.0130	.0286	.0029	aprx.0	.0446
G 63	aprx.0	aprx.0	.0040	.0284	.0029	aprx.0	.0353
G 64	aprx.0	aprx.0	.0012	.0292	.0029	aprx.0	.0322
G 65	aprx.0	aprx.0	.0002	.0300	.0029	aprx.0	.0332
G 66	aprx.0	aprx.0	.0000	.0486	.0057	aprx.0	.0544
G 72	aprx.0	aprx.0	.0234	.0284	.0029	aprx.0	.0545
G 73	aprx.0	aprx.0	.0036	.0272	.0029	aprx.0	.0338
G 74	aprx.0	aprx.0	.0006	.0252	.0029	aprx.0	.0287
G 75	aprx.0	aprx.0	.0000	.0242	.0029	aprx.0	.0271
G 76	aprx.0	aprx.0	aprx.0	.0254	.0029	aprx.0	.0283
G 77	aprx.0	aprx.0	aprx.0	.0410	.0057	aprx.0	.0467

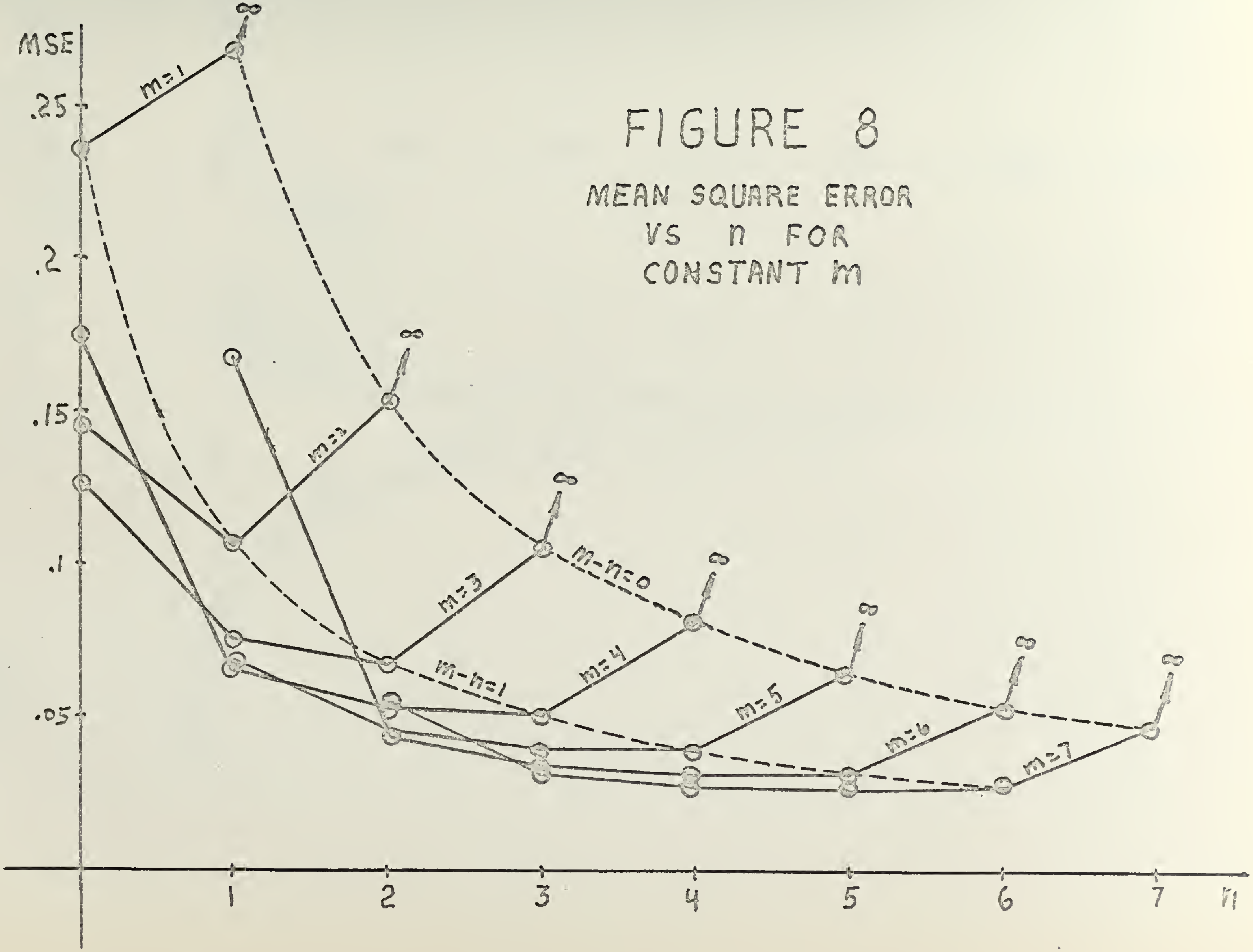
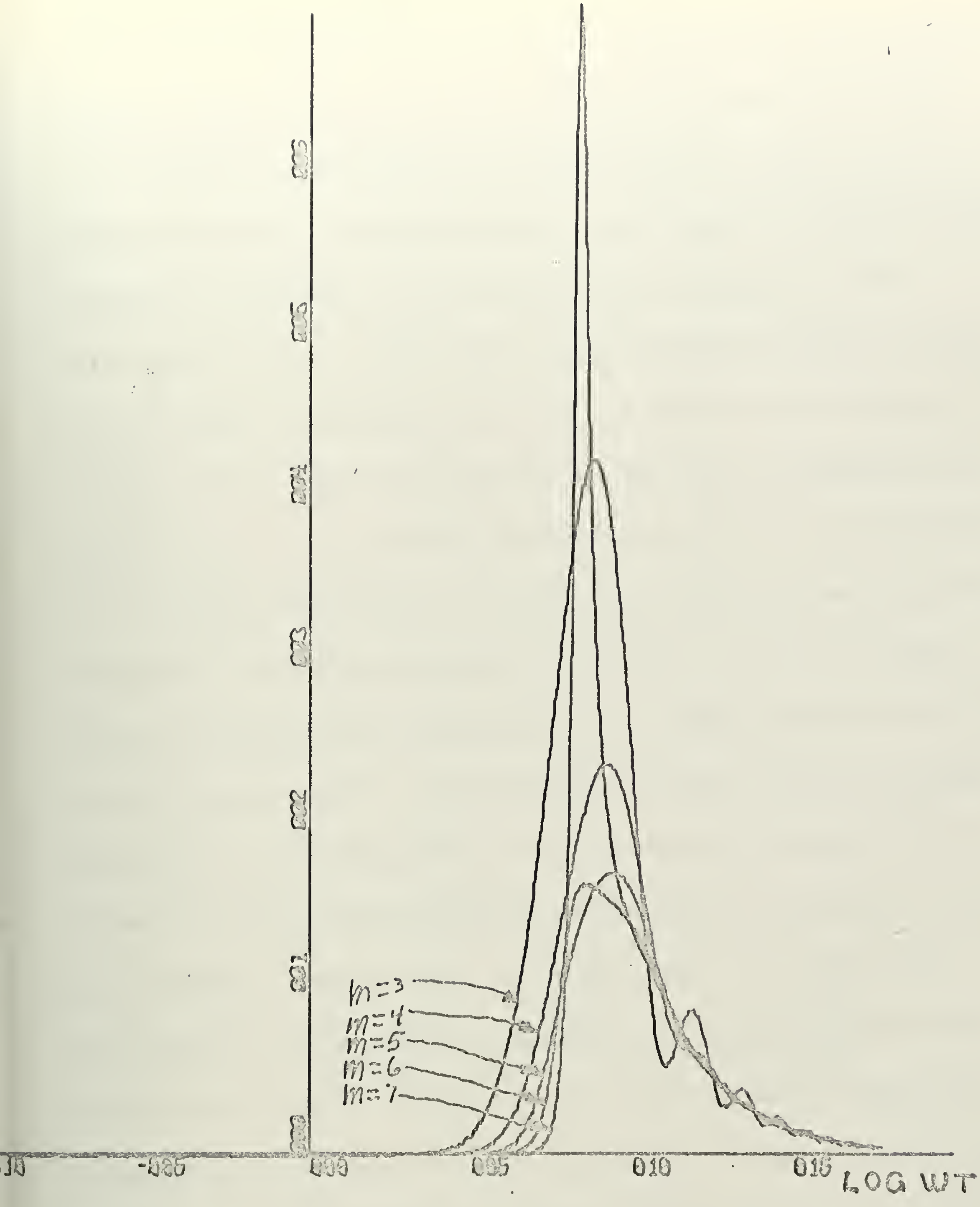


FIGURE 8

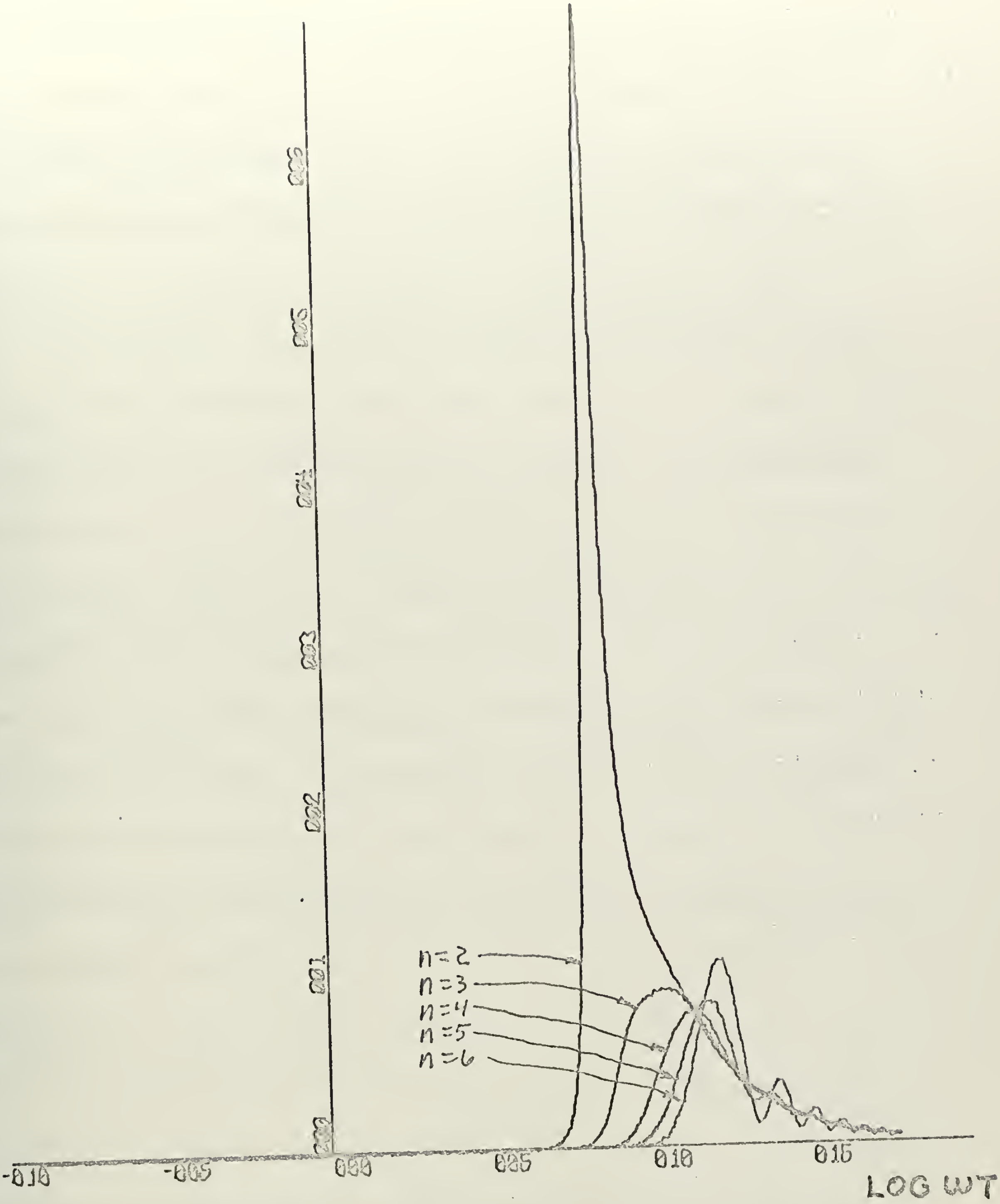
MEAN SQUARE ERROR VS n FOR CONSTANT m



X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E-03 UNITS/INCH.

POWER DENSITY SPECTRUM - G(M2)
PAGE -FIGURE 9(A)

when n becomes larger than m the integral square error goes to infinity. Moving vertically from one solid line to another represents a change in the order of s in the denominator while holding the order of s in the numerator constant. For these changes, the integral square error usually decreases as the order of s is increased. The exceptions to this occur when there is a pole "close" to the imaginary axis (Fig. 6) such as G_{40} and G_{61}. The dotted lines in Fig. 8 represent the change in the integral square error as both indicies are changed, but the difference is held constant. It is readily observed that the amount of improvement in the integral square error is exponentially decreasing as the order (and complexity) of the transfer function is increasing. Figs. 9 (a and b) show the changes in the contribution to the integral square error by the various frequency bands. It is observed, from Fig. 9(a), that changing the order of s in the denominator, while holding the order of s in the numerator constant, has little effect on which band contributes the most to the integral square error. On the other hand, Fig. 9(b) shows that changing n while holding m constant has a considerable effect on the frequency band contributing the most area. As n is increased, the higher frequency bands tend to contribute more. This fact could be used to advantage in many systems. A large number of closed loop systems tend to act as a low pass filter. If one of the Pade' derived filters is used in such a closed loop system, the low pass filtering effect of the system should suppress the higher frequencies that are weighted more by the high pass feedforward filter. This means that the integral square error for a unit step input to such a system would be less than that



X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E-03 UNITS/INCH.
POWER DENSITY SPECTRUM - G(7N)
PAGE -FIGURE 9(B)

when n becomes larger than m the integral square error goes to infinity. Moving vertically from one solid line to another represents a change in the order of s in the denominator while holding the order of s in the numerator constant. For these changes, the integral square error usually decreases as the order of s is increased. The exceptions to this occur when there is a pole "close" to the imaginary axis (Fig. 6) such as G_{40} and G_{61}. The dotted lines in Fig. 8 represent the change in the integral square error as both indicies are changed, but the difference is held constant. It is readily observed that the amount of improvement in the integral square error is exponentially decreasing as the order (and complexity) of the transfer function is increasing. Figs. 9 (a and b) show the changes in the contribution to the integral square error by the various frequency bands. It is observed, from Fig. 9(a), that changing the order of s in the denominator, while holding the order of s in the numerator constant, has little effect on which band contributes the most to the integral square error. On the other hand, Fig. 9(b) shows that changing n while holding m constant has a considerable effect on the frequency band contributing the most area. As n is increased, the higher frequency bands tend to contribute more. This fact could be used to advantage in many systems. A large number of closed loop systems tend to act as a low pass filter. If one of the Pade' derived filters is used in such a closed loop system, the low pass filtering effect of the system should suppress the higher frequencies that are weighted more by the high pass feedforward filter. This means that the integral square error for a unit step input to such a system would be less than that

calculated above. It would therefore be advantageous to use the filter transfer function with the highest frequency weighting in systems that act as low pass filters.

The Integral Square Error Index.(Time Domain). In order to check the above results, the integral square error is calculated for a few cases in the time domain. The same hypothetical experiment, shown in Fig. 7, is used. The equation for the error signal is

$$Y(s) = \frac{(1 - e^{-sT} - G_{mn}(s))}{s} \tag{28}$$

Taking the inverse LaPlace Transform

$$Y(\tau) = U(\tau) - U(\tau - T) - g_1(\tau) \tag{29}$$

where

$$\begin{aligned} U(\tau - T) &= 0 \quad \text{IF} \quad \tau - T < 0 \\ &= 1 \quad \text{IF} \quad \tau - T > 0 \end{aligned} \tag{30}$$

and

$$g_1(\tau) = \mathcal{L}^{-1}\left[\frac{G_{mn}(s)}{s}\right] \tag{31}$$

The integral square error is given by the equation

$$\begin{aligned} I = \int_{-\infty}^{\infty} Y^2(\tau)\,d\tau &= \int_{-\infty}^{\infty} U^2(\tau) + U^2(\tau - T) - 2U(\tau)U(\tau - T)\,d\tau \\ &\int_{-\infty}^{\infty} g_1(\tau)\,U(\tau) - g_1(\tau)\,U(\tau - T)\,d\tau \\ &\int_{-\infty}^{\infty} g_1^{2}(\tau)\,d\tau \end{aligned} \tag{32}$$

It is easily seen that the first integral reduces to T (the area under a pulse of magnitude 1 and duration T). The second integral is equal to the area under the response curve of the Pade' derived transfer function, G_{mn}, from time 0 to T. The last integral represents the area under the square of the filter response curve for all time. The lower limits can be changed to 0 since the response does not start prior to the application of the input at time 0. The equation for the integral square error, for a unit step input, therefore reduces to

$$I = T - 2\int_0^T g_1(\tau)\,d\tau + \int_0^\infty g_1^2(\tau)\,d\tau \tag{33}$$

The digital computer program presented in appendix 2(d) was written to compute I from (33). Because of the long computation time, only a few of the Pade' derived filter transfer functions were tested by this method. The results are given in table 2. Since all of the approximations contain decaying exponential terms, the upper limit for the third integral in (33) was set at 20 times the longest time constant. By comparing tables 1 and 2, it is seen that the results of the two methods agree to within the third decimal place. Better agreement can be obtained if the tolerance on the numerical integration is made smaller. Such action would increase the computation time however.

Other Approximations for $1-e^{-sT}$. Several other sets of approximations were investigated but none performed as well as the Pade'. The defining equations for the other approximations investigated are[8]

$$B_n(s) = 1 - \frac{1}{\left(1 + \frac{A}{n}s\right)^n} \tag{34}$$

Table 2

Integral Square Errors (Time Domain) for Several

Pade' Transfer Functions - Step Input

Time Delay - 1 Second

	first integral	second integral	third integral	I.S.E.
G 10	1.	.6321	.5000	.2358
G 11	1.	.8647	1.0000	.2707
G 20	1.	.8006	.7490	.1477
G 21	1.	.8818	.8696	.1059
G 22	1.	.9225	.9992	.1543

Table 3

Integral Square Errors for

B_1 to B_5 and C_1 to C_5

Step Input - Time Delay 1 Second

—

Band designations are the same as for Table 1

—

band	0	1	2	3	4	5	M.S.E.
B 1	aprx.0	.0166	.1868	.0293	.0029	aprx.0	.2355
B 2	aprx.0	.0054	.1289	.0286	.0029	aprx.0	.1662
B 3	aprx.0	.0025	.1012	.0284	.0029	aprx.0	.1353
B 4	aprx.0	.0016	.0840	.0286	.0029	aprx.0	.1168
B 5	aprx.0	.0010	.0719	.0286	.0029	aprx.0	.1044
C 1	aprx.0	.1248	.1808	.0608	.0057	aprx.0	.4676
C 2	aprx.0	.2779	.2852	.0512	.0057	aprx.0	.6201
C 3	aprx.0	.2928	.4262	.0633	.0057	aprx.0	.7884
C 4	aprx.0	.2986	.5621	.0589	.0057	aprx.0	.9253
C 5	aprx.0	.3014	.6621	.0484	.0057	aprx.0	1.0176

and

$$C_N(s) = 1 - \frac{\left(1 - \frac{A}{N} s\right)^N}{\left(1 + \frac{A}{N} s\right)^N} \qquad (35)$$

These were chosen mainly on the basis that they are relatively easy to synthesize. For example, (35) is a lattice network yielding the proper time delay. The results of the investigation are given in table 3. The format is the same as that for table 1 (i.e. the frequencies in each band are the same as for the bands in table 1). It is noted that the integral square error for the B_n approximation becomes smaller as n is increased. However, for the same degree of complexity, the Pade' approximation has a smaller mean square error. The C_n approximation does not perform well at all. The smallest integral square error occurs when n = 1 and as n is increased, the integral square error increases. The smallest integral square error for a C_n approximation is larger than the largest integral square error for a Pade' approximation.

Summary. From the above observations, it appears that the Pade' approximation derived from the largest number of terms in the power series has the smallest integral square error and is therefore the best approximation to the ideal transfer function, $1-e^{-sT}$. This statement is, of course, subject to the condition that the denominator is of higher order than the numerator and that there are no poles in the right half of the s plane. It is speculated however, that the use of this approximation may not be the most practical in many cases. The increase in complexity (and therefore in difficulty in synthesis) may far outweigh the advantage of a small decrease in the integral square error. As stated before, the

selection of how complex a Pade' derived filter transfer function to use will depend on the system being compensated, the duration of the time delay, the nature of the input signal, etc. The results of this chapter can be summarized as follows:

1. The Pade' approximation is a method for representing a transcendental function as a rational fraction of polynomials. It is found by the simultaneous solution of a series of algebraic equations.

2. Some of the Pade' approximations of the desired function $1-e^{-sT}$ can be realized with lumped, linear passive elements.

3. The best approach in choosing a filter transfer function is through the frequency domain. In this procedure, a quantity called the Spectral Weighting Function is generated from the desired transfer function and the transfer function of a filter under consideration. The spectral weighting function graphically shows which frequencies in the input signal contribute most to the integral square error of the filter. When the spectral weighting function is multiplied by the power density spectrum of an input signal, the power density spectrum of the error is obtained. The filter with the minimum area under the power density spectrum of the error is the one with the minimum integral square error and is considered "the best."

4. The Pade' approximations have the smallest integral square error of all of the approximations investigated for the step input.

IV. EXPERIMENTAL INVESTIGATIONS

In order to examine the effects of approximation in the feedforward compensation schemes outlined earlier, a hybrid computer model was tested with several of the Pade' derived filters in the feedforward path. The specific objectives of the tests were to ascertain the derogatory effects of using an approximation, rather than the ideal transfer function $1-e^{-sT}$, and to determine what effects the order of the approximation has on the time domain performance of the feedforward filter.

The Plant. For simplicity, the plant described in the introduction was the one used for the hybrid computer model. The plant is shown again in Fig. 10(a). Fig. 10(b) shows the computer model. An EAC model TR-20 analog computer was used to simulate the linear portions of the plant. A CDC model 160 digital computer was used as the plant time delay element. In addition, a portion of the analog computer provided signal conditioning for proper A/D and D/A conversion. The A/D converter requires all signal inputs to be between 0 and -10 volts. The analog computer setup is shown in Fig. 11. The maximum allowable signal at any point in the computer model is chosen to be ±8 volts. This is a convenient choice of limits since the digital computer operates on the octal number system and a direct correlation between voltage levels in the analog computer and numbers in the digital computer memory results.

The Digital Time Delay. The basic idea behind the digital time delay is not complex. The analog computer output is sampled periodically

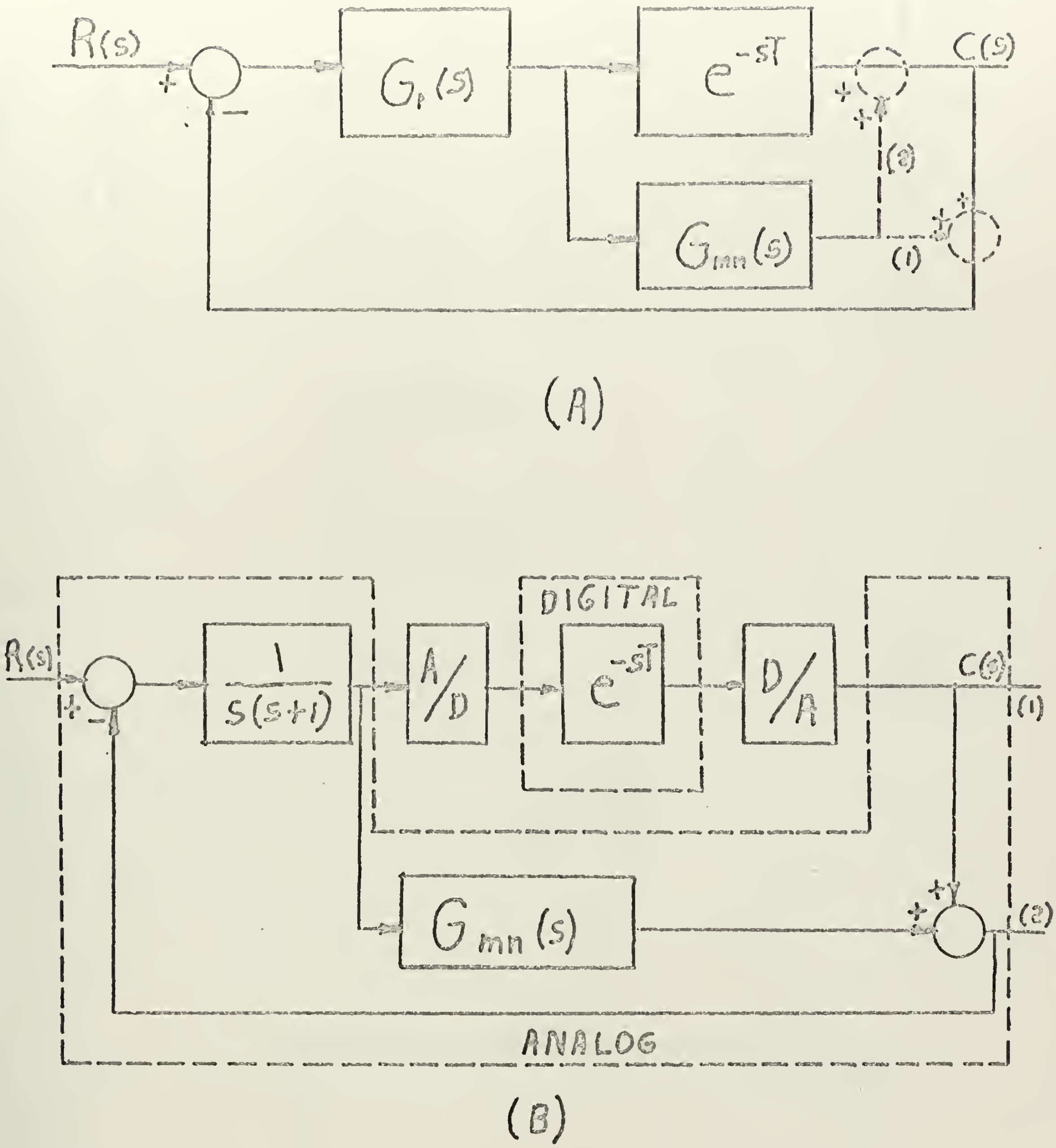


FIGURE 10

PLANT SIMULATION

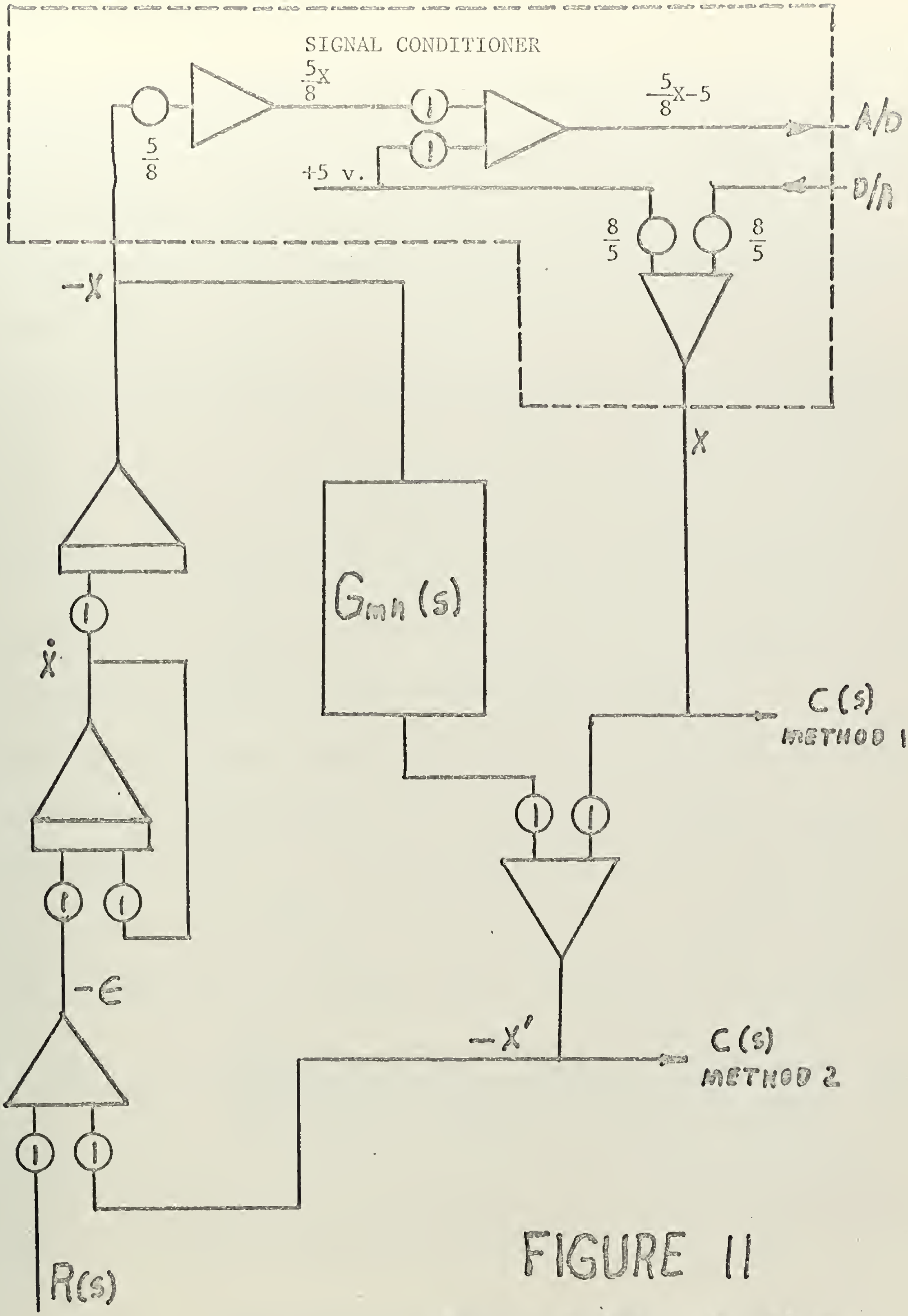


FIGURE 11

ANALOG PLANT SIMULATION

and the number corresponding to the amplitude of that sample is stored in the computer memory. At the end of the time delay period T, the number is returned to the analog computer as a voltage signal through a hold element. If the sampling rate is fast enough, the voltage returned to the analog computer appears to be a continuous time varying signal. A flow graph of this operation is shown in Fig. 12. The computer program written to perform this operation is given in appendix 2(f). There is a difficulty in the use of the program listed in appendix 2(f) however. As the time delay is increased, the sampling rate is decreased. This program is only adequate for for delays of several minutes or less. For time delays longer than this, the output of the D/A converter is better described as piecewise constant rather than continuous.

Simulation of a Typical Pade' Derived Filter. The most difficult step in setting up a computer model of the system is the simulation of the feedforward filter, G_{mn}. Large gains (larger than 10^7 in G_{77}) are required in all but the simplest of the Pade' derived filters. The steps taken to simulate G_{43} are given here as an example of the procedure.

$$G_{43} = \frac{840sT + 60(sT)^2 + 20(sT)^3 + (sT)^4}{840 + 480sT + 120(sT)^2 + 16(sT)^3 + (sT)^4} \tag{36}$$

Dividing the numerator and denominator of (36) by $(sT)^4$ and setting equal to the ratio of the output to the input.

$$\frac{\Theta_o}{\Theta_{in}} = \frac{840\frac{1}{(sT)^3} + 60\frac{1}{(sT)^2} + 20\frac{1}{(sT)} + 1}{840\frac{1}{(sT)^4} + 480\frac{1}{(sT)^3} + 120\frac{1}{(sT)^2} + 16\frac{1}{sT} + 1} \tag{37}$$

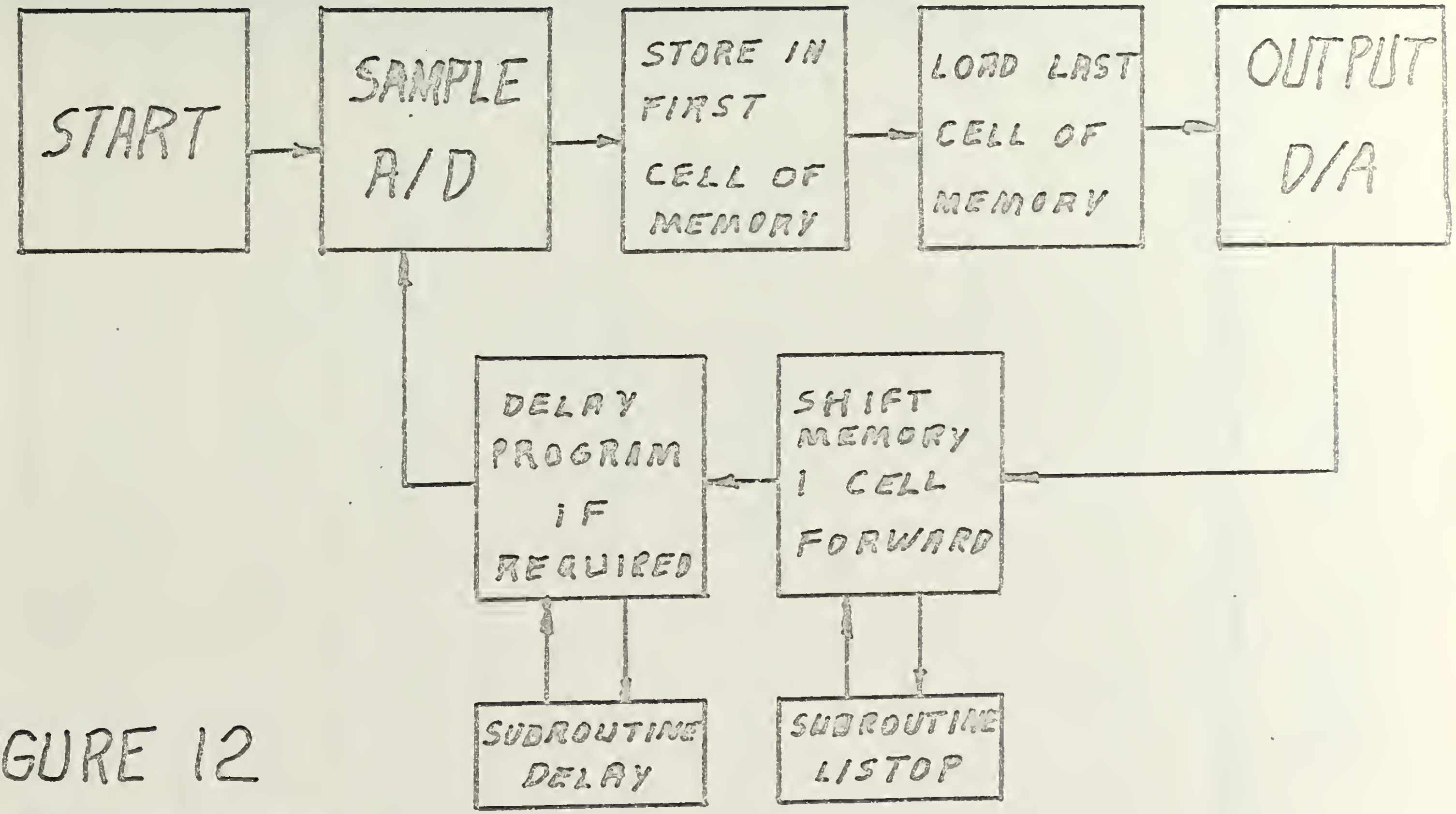


FIGURE 12

DIGITAL TIME DELAY

By cross multiplication and rearranging terms

$$\begin{aligned}\Theta_o = {} & \Theta_{in}\left[1 + 20\frac{1}{ST} + 60\frac{1}{(ST)^2} + 840\frac{1}{(ST)^3}\right] \\ & - \Theta_o\left[16\frac{1}{ST} + 120\frac{1}{(ST)^2} + 480\frac{1}{(ST)^3} + 840\frac{1}{(ST)^4}\right]\end{aligned} \tag{38}$$

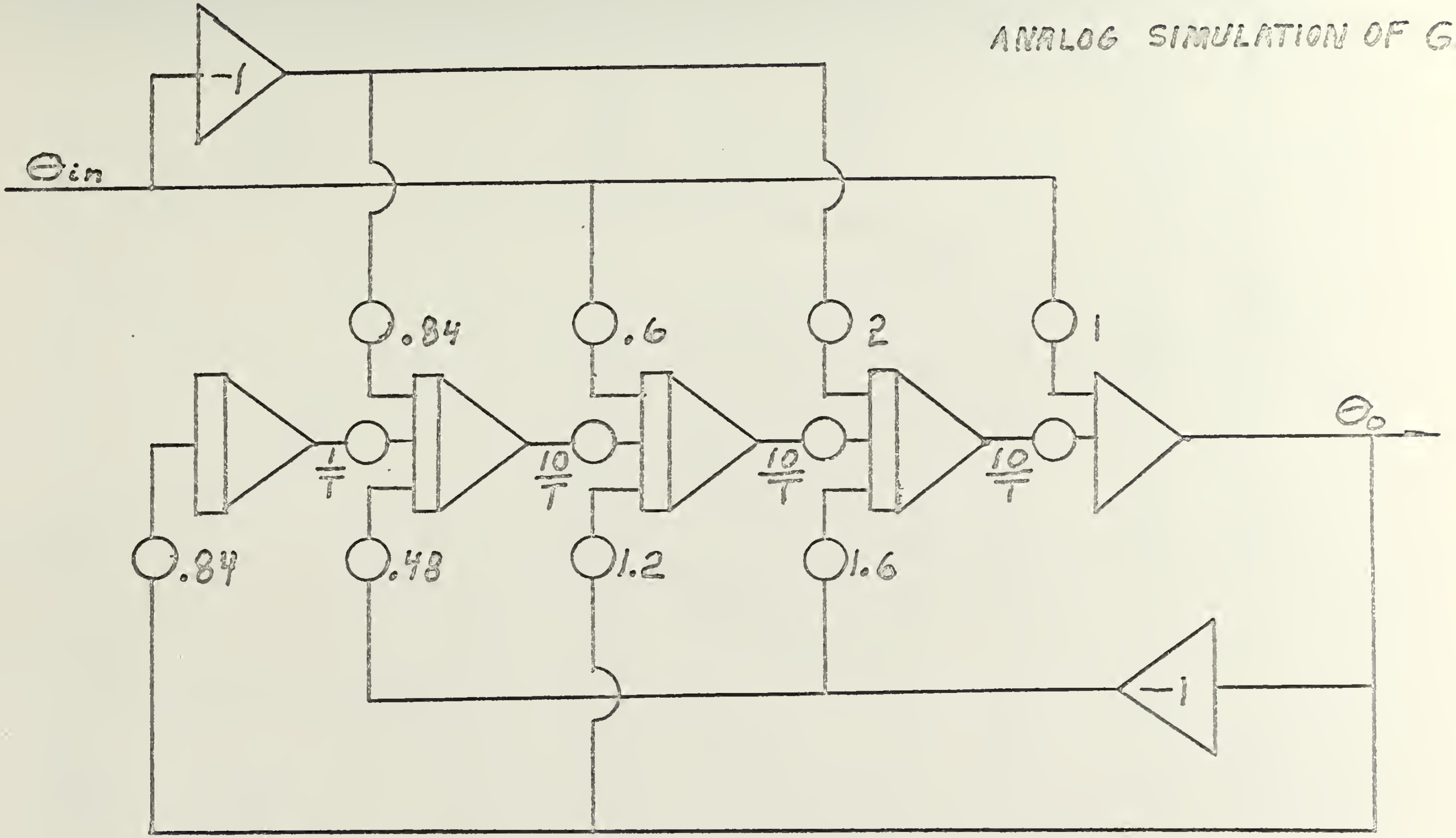
(38) is in the proper form for the simulation on the analog computer. The analog computer setup for G_{43} is shown in Fig. 13. The other Pade' derived filter approximations are simulated in the same way.

The Step Response. It was stated in the introduction that the system under consideration was stable with a time delay of 1 second but unstable if the time delay was increased to 2 seconds. The Bode diagram for the plant was presented in Fig. 2 to support this. To verify this, the feedforward filter was disconnected in the computer model and the model was subjected to a 1 volt step. The results are shown in Fig. 14. The plant is definitely unstable for a time delay of 2 seconds.

To meet the objectives of the tests, three of the Pade' derived filter transfer functions were used as the feedforward network in the computer model. The three that were used are G_{21}, G_{43}, and G_{65}. These were chosen because they had the minimum integral square error for a given value of m. The time delay was set to 2, 3, and 4 seconds. At all three of the time delay settings, the plant without compensation would be unstable. The results of the tests, for both compensation methods, are shown in Figs. 15 (a, b and c). It is observed that all three Pade' derived filter transfer functions stabilize the system for all three values of time delay tested. For the longer delays, the more

FIGURE 13

ANALOG SIMULATION OF G_{43}



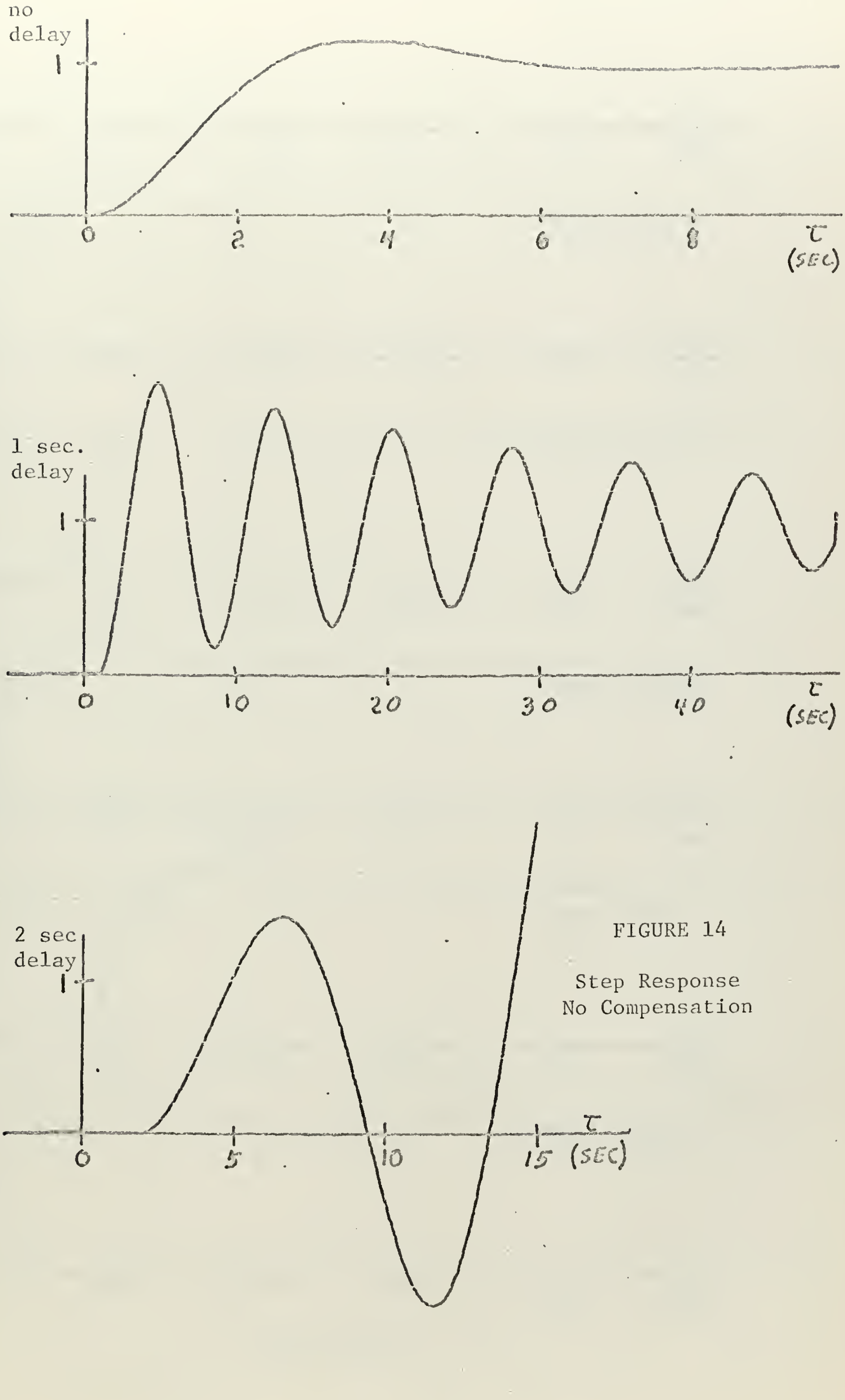


FIGURE 14

Step Response
No Compensation

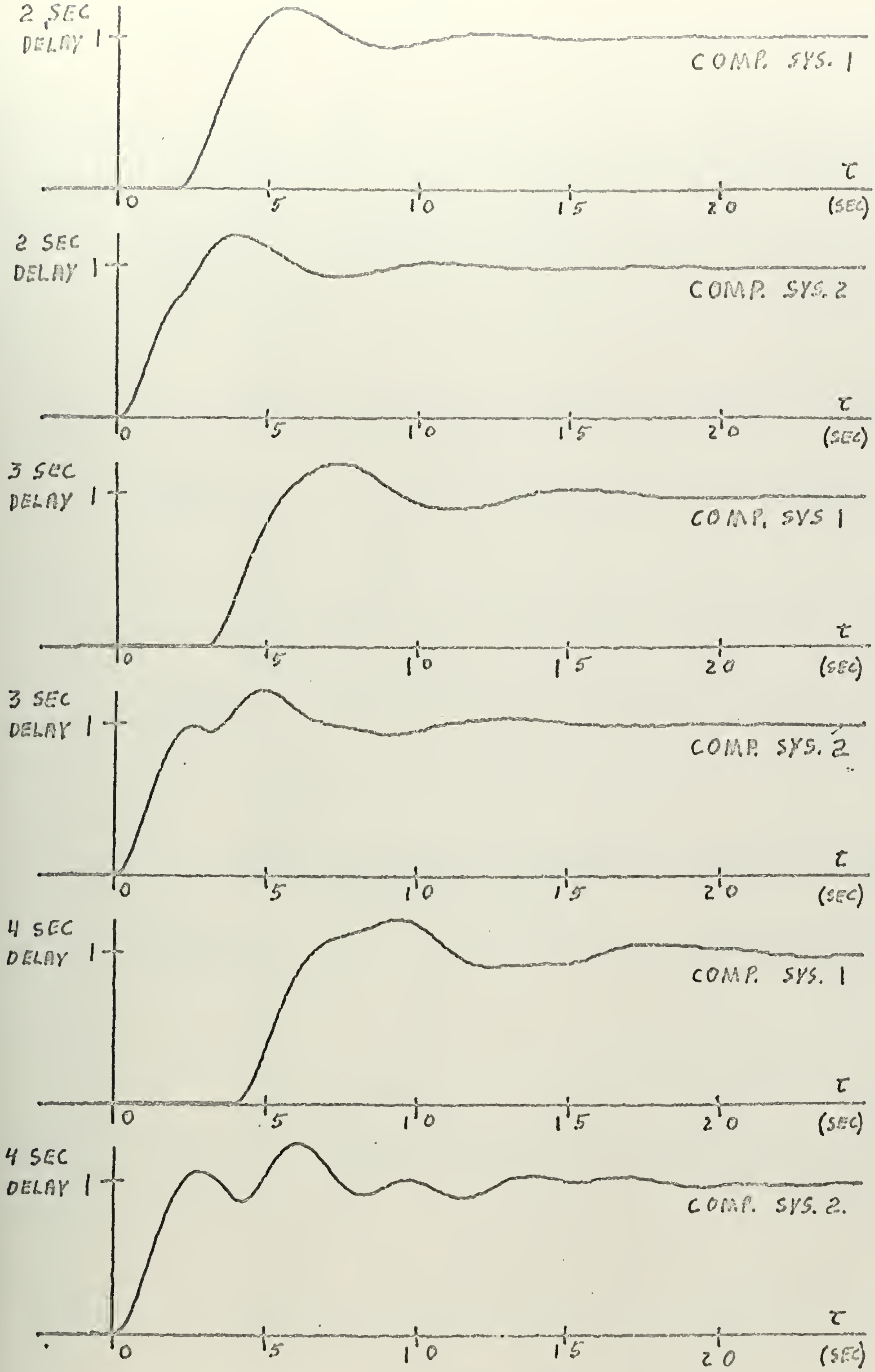


FIGURE 15(a)

Step Response with G_{21} Compensator

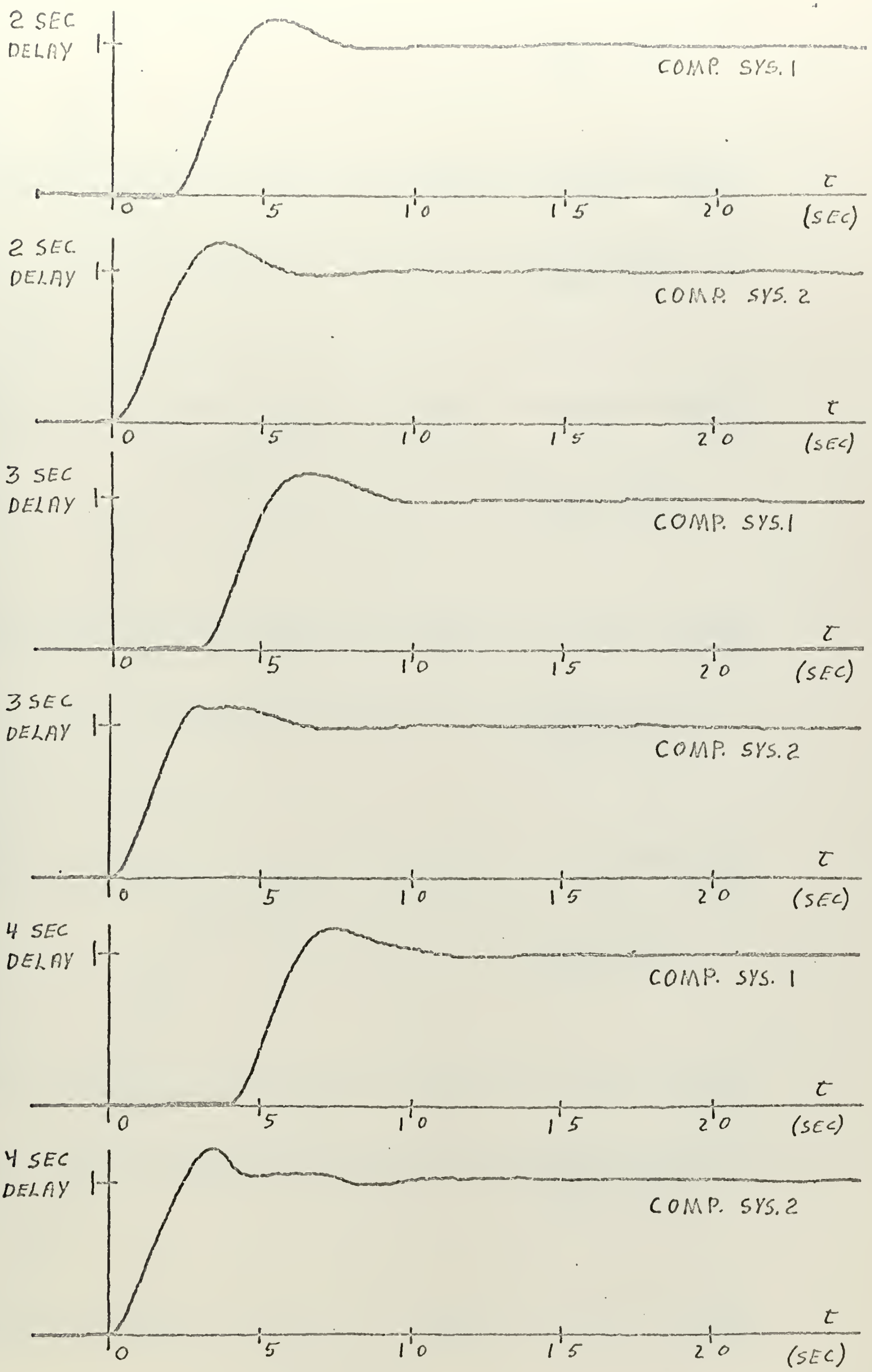


FIGURE 15(b)
Step Response with G_{43} Compensator

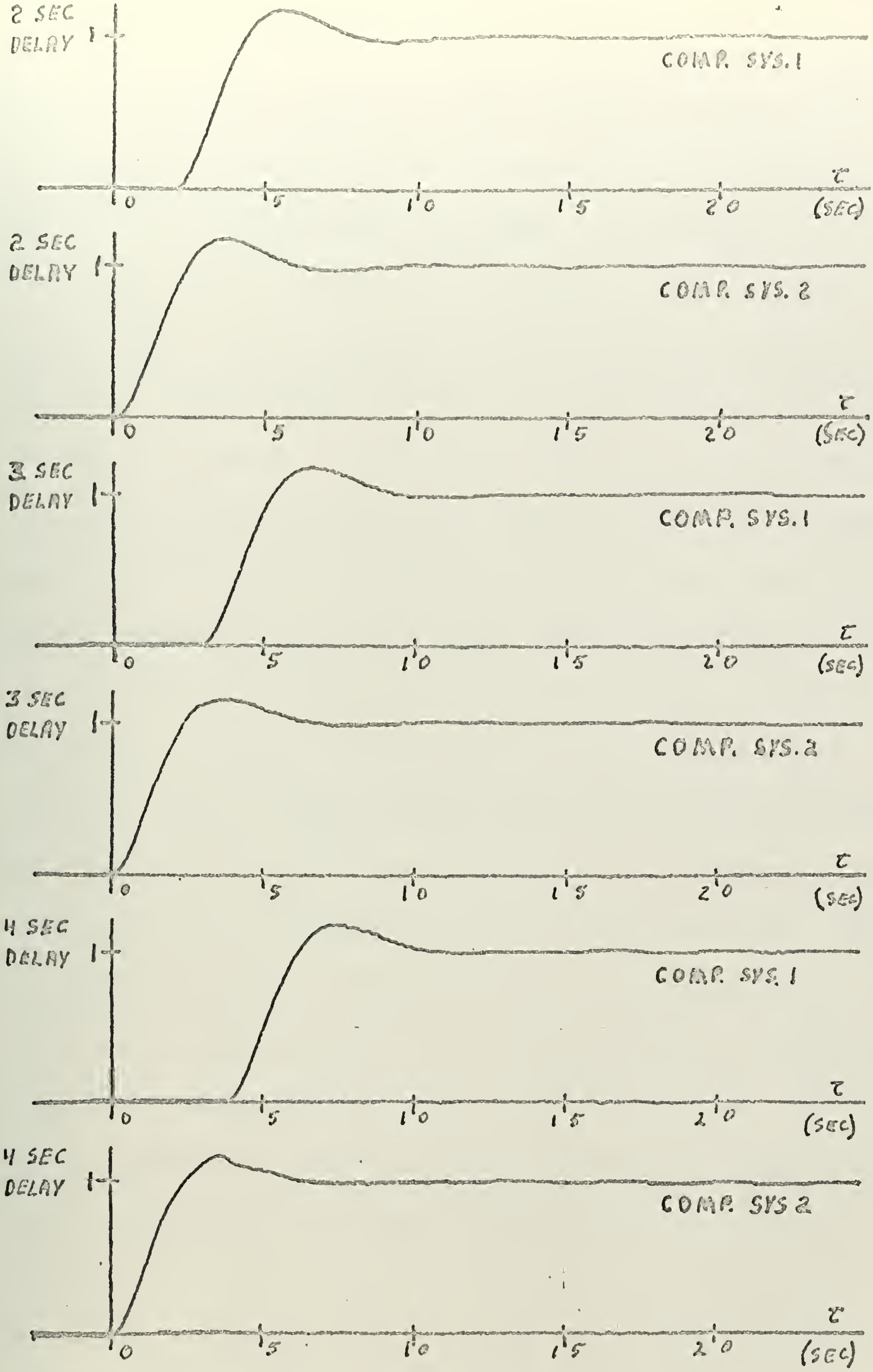


FIGURE 15(c)

Step Response with G_{65} Compensator

complex filters do a better job with respect to the shape of the response curve. The response with the G_{65} filter and with a 4 second time delay looks just like the response of the uncompensated system with no time delay, with the exception of the time shift. It is interesting to note that the two compensation methods (Fig. 4) do not give the same compensated system output. The best example of observing this is the response of G_{21} with a 4 second delay, Fig. 15(a). The response due to compensation method 2 oscillates much more than the response due to compensation method 1 and the settling time is longer. On the whole, it appears that compensation method 1 performs better than compensation method 2. This is very fortunate since it was noted earlier that practical considerations might require the use of this compensation scheme.

Sensitivity. No extensive investigation was made into the area of sensitivity. Only one short experiment was performed to get an idea of the tolerance limits on the filter time delay setting. The plant was simulated on the hybrid computer with a two second time delay. This is an unstable condition. The plant was then compensated, using compensation method 1 and the Pade' derived filter G_{21}. The filter time delay was varied from 1 to 5 seconds, in half second steps. The 1 second setting was the shortest delay that could be conveniently set on the computer and at 5 seconds, the system was at the margin of stability. The results of some of the other trials are shown in Fig. 16. It is observed that the system is stable for a large range of settings of filter time delay. The quality of the response deteriorates as the difference between the plant and filter time delays increases but the

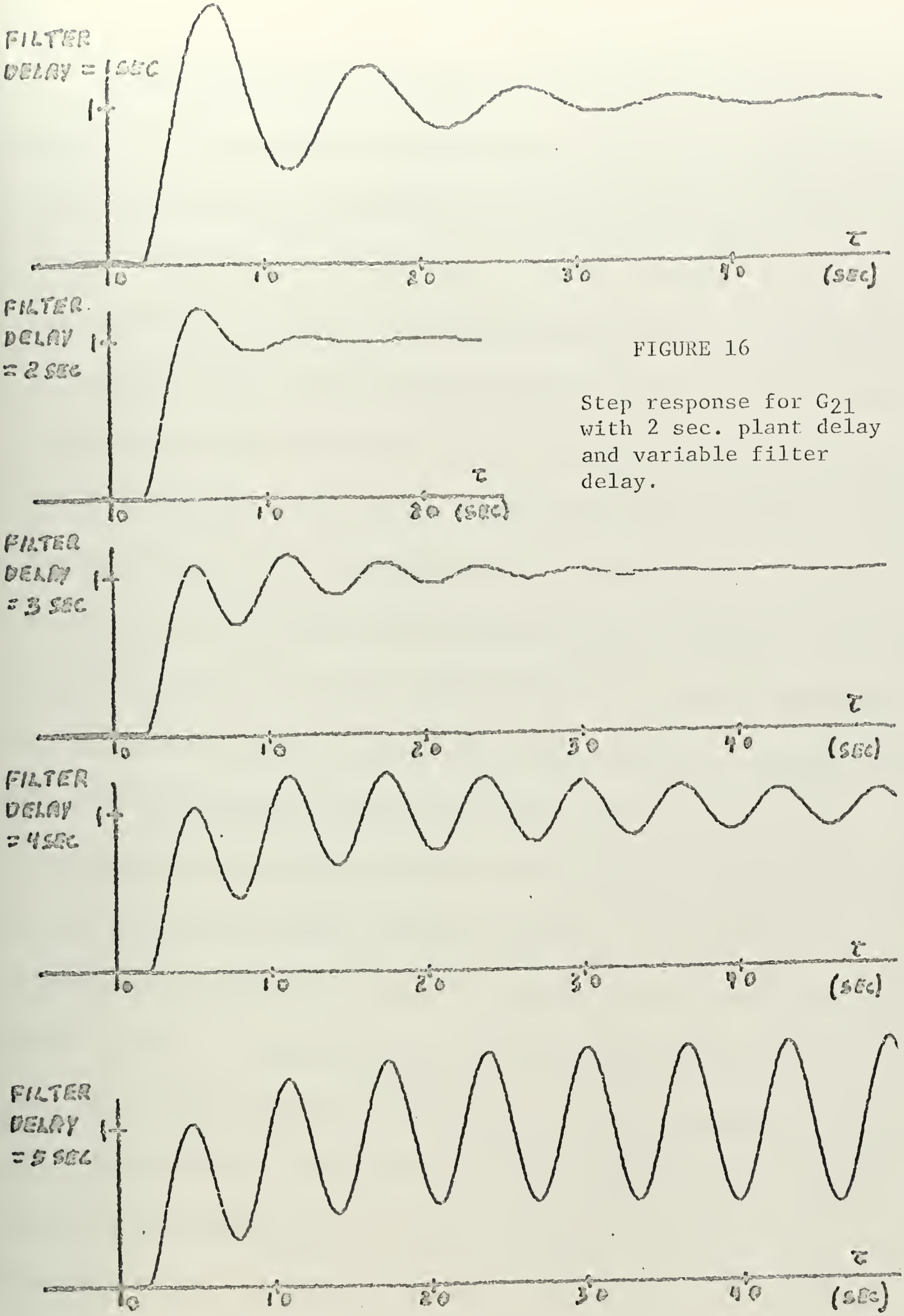


FIGURE 16

Step response for G_{21} with 2 sec. plant delay and variable filter delay.

V. CONCLUSION

It was easily shown, in Chapter 2 of this thesis, that the feedforward compensation techniques presented in Fig. 4 would remove the phase lag due to a time delay in a closed loop control system and would therefore improve the stability of such a system. It was then implied that the required feedforward network could not be synthesized with lumped, linear, passive, time invarient circuit elements. The problem was therefore reduced to one of finding a rational approximation that would perform as near to the unrealizable ideal circuit as possible. Since the function to be synthesized is a common one, a number of approximations and approximation techniques were available to choose from. The performance index, chosen to rate the various approximations, was the minimum integral square error between the step response of the ideal and approximation filters. The Pade' method of polynomial approximation was finally used since it was simple to apply and it gave a wide variation in transfer function complexity. This large variety of circuits means that a large range of integral square error was available to test the effects of approximation in the two compensation schemes. It was found, through computer simulations, that approximations of the ideal transfer function effectively stabilize systems with time delay. Even poor approximations, i.e. filters with large integral square errors, were able to effect a degree of stability in such systems. In addition, it was found that the stabilizing properties of the two compensation schemes were not affected by small changes in the time delay setting

system remains stable. This feature is useful in situations where the plant time delay is either not known exactly or it changes.

Summary. The observations made in this chapter can be summarized as follows:

1. The Pade' derived filter transfer functions make effective feedforward filters in the compensation schemes outlined in Fig. 4. As anticipated, the more complex transfer functions, which are better approximations to the ideal transfer function, give a better system response.

2. Compensation method 1 (Fig. 4(a)) gives a better system response than compensation method 2.

3. The ability of the Pade' derived filter transfer functions to stabilize an unstable plant is not very sensitive to changes in the filter time delay setting.

V. CONCLUSION

It was easily shown, in Chapter 2 of this thesis, that the feedforward compensation techniques presented in Fig. 4 would remove the phase lag due to a time delay in a closed loop control system and would therefore improve the stability of such a system. It was then implied that the required feedforward network could not be synthesized with lumped, linear, passive, time invarient circuit elements. The problem was therefore reduced to one of finding a rational approximation that would perform as near to the unrealizable ideal circuit as possible. Since the function to be synthesized is a common one, a number of approximations and approximation techniques were available to choose from. The performance index, chosen to rate the various approximations, was the minimum integral square error between the step response of the ideal and approximation filters. The Pade' method of polynomial approximation was finally used since it was simple to apply and it gave a wide variation in transfer function complexity. This large variety of circuits means that a large range of integral square error was available to test the effects of approximation in the two compensation schemes. It was found, through computer simulations, that approximations of the ideal transfer function effectively stabilize systems with time delay. Even poor approximations, i.e. filters with large integral square errors, were able to effect a degree of stability in such systems. In addition, it was found that the stabilizing properties of the two compensation schemes were not affected by small changes in the time delay setting

of the approximation feedforward filter. In general, it can be concluded that both the compensation methods presented in Chapter 2 are excellent compensators for time delay and that the required feedforward circuits are not very sensitive to approximation effects.

The concept of a frequency domain analysis of the minimum integral square error criterion, as developed in Chapter 3, is as important as the idea of feedforward compensation. In most of the literature, the time domain analysis is used. While the time domain technique gives satisfactory answers, i.e. gives the integral square error for a problem under consideration, the frequency domain analysis gives this information and much more. The spectral weighting function, an intermediate step in the frequency domain analysis, gives the user a graphical indication of the frequencies that are contributing to integral square error. This means that the user can find a solution for his problem based on the frequency spectrum of the input signal and the spectral weighting functions of the various solutions under consideration. The spectral weighting function can be computed once and used to calculate the integral square error for any input. These advantages are not available when the time domain analysis is used.

Topics for Further Study. The ideas used in this thesis can be projected into additional areas of study. For example, work can be done in the case where the time delay cannot be isolated from the remainder of the open loop transfer function of the plant. Also, the idea of a sensitivity investigation could receive additional attention. It is interesting to note that the ideal feedforward transfer function is very similar to the

transfer function of a Zero Order Hold circuit, $(1-e^{-sT})/s$. A number of the Pade' derived filters found in appendix 1 could possibly be used in conjunction with an integrator to produce a passive approximation of a zero order hold circuit. The Pade' polynomials are a promising method for finding a rational approximation for any irrational function and work can be done in the implementation of such functions. Another area where work can be done is to consider phase relationships in the frequency domain analysis of the error. Only the amplitude relationship (integral square error) was considered in this thesis. It would be interesting to see what additional information could be obtained from such an analysis.

BIBLIOGRAPHY

(1) Buckley, P. S., PROCEEDINGS OF THE 1st INTERNATIONAL CONGRESS OF THE INTERNATIONAL FEDERATION OF AUTOMATIC CONTROL, Moscow, 1960, Volume 1, pp 33-39.

(2) Buckley, P. S., TECHNIQUES OF PROCESS CONTROL, John Wiley and Sons, New York, 1964, pp 92-95.

(3) Stewart, J. L., GENERALIZED PADE' APPROXIMATION, Proceedings of the Institute of Radio Engineers, Volume 48, December 1960, pp 2003-2008.

(4) Karni, Shlomo, NETWORK THEORY; ANALYSIS AND SYNTHESIS, Allyn and Bacon, Boston, 1966, pp 367-374.

(5) Truxal, John C., CONTROL SYSTEM SYNTHESIS, McGraw-Hill Book Company, New York, 1955, pp 550.

(6) Van Valkenburg, M. E., INTRODUCTION TO MODERN NETWORK SYNTHESIS, John Wiley and Sons, New York, 1960, pp 248-288.

(7) Newton, G. C., L. A. Gould and J. F. Kaiser, ANALYTICAL DESIGN OF LINEAR FEEDBACK CONTROLS, John Wiley and Sons, New York, 1957, pp 109-125.

(8) Hazony, D., ELEMENTS OF NETWORK SYNTHESIS, Reinholt Publishing Company, New York, 1963, pp 174-175.

APPENDIX 1(A)

PADE' APPROXIMATION OF e^{-sT}

$X = sT$

$$E_{00} = 1$$

$$E_{01} = 1-X$$

$$E_{02} = \frac{2-2X+X^2}{2}$$

$$E_{03} = \frac{6-6X+3X^2-X^3}{6}$$

$$E_{04} = \frac{24-24X+12X^2-4X^3+X^4}{24}$$

$$E_{10} = \frac{1}{1+X}$$

$$E_{11} = \frac{2-X}{2+X}$$

$$E_{12} = \frac{6-4X+X^2}{6+2X}$$

$$E_{13} = \frac{24-18X+6X^2-X^3}{24+6X}$$

$$E_{14} = \frac{120-96X+36X^2-8X^3+X^4}{120+24X}$$

$$E_{20} = \frac{2}{2+2X+X^2}$$

$$E_{21} = \frac{6-2X}{6+4X+X^2}$$

$$E_{22} = \frac{12-6X+X^{2}}{12+6X+X^{2}}$$

$$E_{23} = \frac{60-36X+9X^{2}-X^{3}}{60+24X+3X^{2}}$$

$$E_{24} = \frac{360-240X+72X^{2}-12X^{3}+X^{4}}{360+120X+12X^{2}}$$

$$E_{30} = \frac{6}{6+6X+3X^{2}+X^{3}}$$

$$E_{31} = \frac{24-6X}{24+18X+6X^{2}+X^{3}}$$

$$E_{32} = \frac{60-24X+3X^{22}}{60+36X+9X^{2}+X^{3}}$$

$$E_{33} = \frac{120-60X+12X^{2}-X^{3}}{120+60X+12X^{2}+X^{3}}$$

$$E_{34} = \frac{840-480X+12X^{2}-16X^{3}+X^{4}}{840+360X+60X^{2}+4X^{3}}$$

$$E_{40} = \frac{24}{24+24X+12X^{2}+4X^{3}+X^{4}}$$

$$E_{41} = \frac{120-24X}{120+96X+36X^{2}+8X^{3}+X^{4}}$$

$$E_{42} = \frac{360-120X+12X^{2}}{360+240X+72X^{2}+12X^{3}+X^{4}}$$

$$E_{43} = \frac{840-360X+60X^{2}-4X^{3}}{840+480X+120X^{2}+16X^{3}+X^{4}}$$

$$E_{44} = \frac{1680-840X+180X^{2}-20X^{3}+X^{4}}{1680+840X+180X^{2}+20X^{3}+X^{4}}$$

$$E_{50} = \frac{120}{120+120X+60X^{2}+20X^{3}+5X^{4}+X^{5}}$$

$$E_{51} = \frac{720-120X}{720+600X+240X^{2}+60X^{3}+10X^{4}+X^{5}}$$

$$E_{52} = \frac{2520-720X+60X^{2}}{2520+1800X+600X^{2}+120X^{3}+15X^{4}+X^{5}}$$

$$E_{53} = \frac{6720-2520X+360X^{2}-20X^{3}}{6720+4200X+1200X^{2}+200X^{3}+20X^{4}+X^{5}}$$

$$E_{54} = \frac{15120-6720X+1260X^{2}-120X^{3}+5X^{4}}{15120+8400X+2100X^{2}+300X^{3}+25X^{4}+X^{5}}$$

$$E_{55} = \frac{30240-15120X+3360X^{2}-420X^{3}+30X^{4}-X^{5}}{30240+15120X+3360X^{2}+420X^{3}+30X^{4}+X^{5}}$$

$$E_{60} = \frac{720}{720+720X+360X^{2}+120X^{3}+30X^{4}+6X^{5}+X^{6}}$$

$$E_{61} = \frac{5040-720X}{5040+4320X+1800X^{2}+480X^{3}+90X^{4}+12X^{5}+X^{6}}$$

$$E_{62} = \frac{20160-5040X+360X^{2}}{20160+15120X+5400X^{2}+1200X^{3}+180X^{4}+18X^{5}+X^{6}}$$

$$E_{63} = \frac{60480-20160X+2520X^{2}-120X^{3}}{60480+40320X+12600X^{2}+2400X^{3}+300X^{4}+24X^{5}+X^{6}}$$

$$E_{64} = \frac{151200-60480X+10080X^{2}-840X^{3}+30X^{4}}{151200+90720X+25200X^{2}+14200X^{3}+450X^{4}+30X^{5}+X^{6}}$$

$$E_{65} = \frac{332640-151200X+30240X^{2}-3360X^{3}+210X^{4}-6X^{5}}{332640+181440X+45360X^{2}+6720X^{3}+630X^{4}+36X^{5}+X^{6}}$$

$$E_{66} = \frac{665280-332640X+75600X^{2}-10080X^{3}+840X^{4}-42X^{5}+X^{6}}{665280+332640X+75600X^{2}+10080X^{3}+840X^{4}+42X^{5}+X^{6}}$$

$$E_{70} = \frac{5040}{5040+5040X+2520X^{2}+840X^{3}+210X^{4}+42X^{5}+7X^{6}+X^{7}}$$

$$E_{71} = \frac{40320-5040X}{40320+35280X+15120X^{2}+4200X^{3}+840X^{4}+126X^{5}+14X^{6}+X^{7}}$$

$$E_{72} = \frac{181440-40320X+2520X^{2}}{181440+141120X+52920X^{2}+12600X^{3}+2100X^{4}+252X^{5}+21X^{6}+X^{7}}$$

$$E_{73} = \frac{604800-181440X+20160X^{2}-840X^{3}}{604800+423360X+141120X^{2}+29400X^{3}+4200X^{4}+420X^{5}+28X^{6}+X^{7}}$$

$$E_{74} = \frac{1663200-604800X+90720X^{2}-6720X^{3}+210X^{4}}{1663200+1058400X+317520X^{2}+58800X^{3}+7320X^{4}+630X^{5}+35X^{6}+X^{7}}$$

$$E_{75} = \frac{3991680-1663200X+302400X^{2}-30240X^{3}+1680X^{4}-42X^{5}}{3991680+2328480X+635040X^{2}+105840X^{3}+11760X^{4}+882X^{5}+42X^{6}+X^{7}}$$

$$E_{76} = \frac{8648640-3991680X+831600X^{2}-100800X^{3}+7560X^{4}-336X^{5}+7X^{6}}{8648640+4656960X+1164240X^{2}+176400X^{3}+17640X^{4}+1176X^{5}+49X^{6}+X^{7}}$$

$$E_{77} = \frac{17297280-8648640X+1995840X^{2}-277200X^{3}+25200X^{4}-1512X^{5}+56X^{6}-X^{7}}{17297280+8648640X+1995840X^{2}+277200X^{3}+25200X^{4}+1512X^{5}+56X^{6}+X^{7}}$$

APPENDIX 1(B)

PADE' APPROXIMATION OF $1-e^{-sT}$

$G_{mn} = 1 - E_{mn}$ $\qquad$ $X = sT$

$$G_{00} = 0$$

$$G_{01} = X$$

$$G_{02} = \frac{2X-X^2}{2}$$

$$G_{03} = \frac{6X-3X^2+X^3}{6}$$

$$G_{04} = \frac{24X-12X^2+4X^3-X^4}{24}$$

$$G_{10} = \frac{X}{1+X}$$

$$G_{11} = \frac{2X}{2+X}$$

$$G_{12} = \frac{6X-X^2}{6+2X}$$

$$G_{13} = \frac{24X-6X^2+X^3}{24+6X}$$

$$G_{14} = \frac{120X-36X^2+8X^3-X^4}{120+24X}$$

$$G_{20} = \frac{2X+X^2}{2+2X+X^2}$$

$$G_{21} = \frac{6X+X^2}{6+4X+X^2}$$

$$G_{22} = \frac{12X}{12+6X+X^2}$$

$$G_{23} = \frac{60X-6X^2+X^3}{60+24X+3X^2}$$

$$G_{24} = \frac{360X-60X^2+12X^3-X^4}{360+120X+12X^2}$$

$$G_{30} = \frac{6X+3X^2+X^3}{6+6X+3X^2+X^3}$$

$$G_{31} = \frac{24X+6X^2+X^3}{24+18X+6X^2+X^3}$$

$$G_{32} = \frac{60X+6X^2+X^3}{60+36X+9X^2+X^3}$$

$$G_{33} = \frac{120X+2X^3}{120+60X+12X^2+X^3}$$

$$G_{34} = \frac{840X-60X^2+20X^3-X^4}{840+360X+60X^2+4X^3}$$

$$G_{40} = \frac{24X+12X^2+4X^3+X^4}{24+24X+12X^2+4X^3+X^4}$$

$$G_{41} = \frac{120X+36X^2+8X^3+X^4}{120+96X+36X^2+8X^3+X^4}$$

$$G_{42} = \frac{360X+60X^2+12X^3+X^4}{360+240X+72X^2+12X^3+X^4}$$

$$G_{43} = \frac{840X+60X^{2}+20X^{3}+X^{4}}{840+480X+120X^{2}+16X^{3}+X^{4}}$$

$$G_{44} = \frac{1680X+40X^{3}}{1680+840X+180X^{2}+20X^{3}+X^{4}}$$

$$G_{50} = \frac{120X+60X^{2}+20X^{3}+5X^{4}+X^{5}}{120+120X+60X^{2}+20X^{3}+5X^{4}+X^{4}}$$

$$G_{51} = \frac{720X+240X^{2}+60X^{3}+10X^{4}+X^{5}}{720+600X+240X^{2}+60X^{3}+10X^{4}+X^{5}}$$

$$G_{52} = \frac{2520X+540X^{2}+120X^{3}+15X^{4}+X^{5}}{2520+1800X+600X^{2}+120X^{3}+15X^{4}+X^{5}}$$

$$G_{53} = \frac{6720X+840X^{2}+220X^{3}+20X^{4}+X^{5}}{6720+4200X+1200X^{2}+200X^{3}+20X^{4}+X^{5}}$$

$$G_{54} = \frac{15120X+840X^{2}+420X^{3}+20X^{4}+X^{5}}{15120+8400X+2100X^{2}+300X^{3}+25X^{4}+X^{5}}$$

$$G_{55} = \frac{30240X+840X^{3}+2X^{5}}{30240+15120X+3360X^{2}+420X^{3}+30X^{4}+X^{5}}$$

$$G_{60} = \frac{720X+360X^{2}+120X^{3}+30X^{4}+6X^{5}+X^{6}}{720+720X+360X^{2}+120X^{3}+30X^{4}+6X^{5}+X^{6}}$$

$$G_{61} = \frac{5040X+1800X^{2}+480X^{3}+90X^{4}+12X^{5}+X^{6}}{5040+4320X+1800X^{2}+480X^{3}+90X^{4}+12X^{5}+X^{6}}$$

$$G_{62} = \frac{20160X+5040X^{2}+1200X^{3}+180X^{4}+18X^{5}+X^{6}}{20160+15120X+5400X^{2}+1200X^{3}+180X^{4}+18X^{5}+X^{6}}$$

$$G_{63} = \frac{60480X+10080X^2+2520X^3+300X^4+24X^5+X^6}{60480+40320X+12600X^2+2400X^3+300X^4+24X^5+X^6}$$

$$G_{64} = \frac{151200X+15120X^2+5040X^3+420X^4+30X^5+X^6}{151200+90720X+25200X^2+4200X^3+450X^4+30X^5+X^6}$$

$$G_{65} = \frac{332640X+15120X^2+10080X^3+420X^4+42X^5+X^6}{332640+181440X+45360X^2+6720X^3+630X^4+36X^5+X^6}$$

$$G_{66} = \frac{665280X+20160X^3+84X^5}{665280+332640X+75600X^2+10080X^3+840X^4+42X^5+X^6}$$

$$G_{70} = \frac{5040X+2520X^2+840X^3+210X^4+42X^5+7X^6+X^7}{5040+5040X+2520X^2+840X^3+210X^4+42X^5+7X^6+X^7}$$

$$G_{71} = \frac{40320X+15120X^2+4200X^3+840X^4+126X^5+14X^6+X^7}{40320+35280X+15120X^2+4200X^3+840X^4+126X^5+14X^6+X^7}$$

$$G_{72} = \frac{181440X+50400X^2+12600X^3+2100X^4+252X^5+21X^6+X^7}{181440+141120X+52920X^2+12600X^3+2100X^4+252X^5+21X^6+X^7}$$

$$G_{73} = \frac{604800X+120960X^2+30240X^3+4200X^4+420X^5+28X^6+X^7}{604800+423360X+141120X^2+29400X^3+4200X^4+420X^5+28X^6+X^7}$$

$$G_{74} = \frac{1663200X+226800X^2+65520X^3+7110X^4+630X^5+35X^6+X^7}{1663200+1058400X+317520X^2+58800X^3+7320X^4+630X^5+35X^6+X^7}$$

$$G_{75} = \frac{3991680X+332640X^2+136080X^3+10080X^4+924X^5+42X^6+X^7}{3991680+2328480X+635040X^2+105840X^3+11760X^4+882X^5+42X^6+X^7}$$

$$G_{76} = \frac{8648640X+332640X^2+277200X^3+10080X^4+1512X^5+42X^6+X^7}{8648640+4656960X+1164240X^2+176400X^3+17640X^4+1176X^5+49X^6+X^7}$$

$$G_{77} = \frac{17297280X+554400X^{3}+3024X^{5}+2X^{7}}{17297280+8648640X+1995840X^{2}+277200X^{3}+25200X^{4}+1512X^{5}+56X^{6}+X^{7}}$$

APPENDIX 1(c)

PADE' APPROXIMATION OF $1-e^{-sT}$

(Note: all listed complex roots have complex conjugates which are not listed but are represented by the open parenthesis.)

$$X = sT$$

$$G_{00} = 0$$

$$G_{01} = X$$

$$G_{02} = \frac{X(X-2)}{2}$$

$$G_{03} = \frac{X(X-1.5+j1.936)(\,)}{6}$$

$$G_{04} = \frac{X(X-2.785)(X-.607+j2.872)(\,)}{24}$$

$$G_{10} = \frac{X}{X+1}$$

$$G_{11} = \frac{2X}{X+2}$$

$$G_{12} = \frac{X(X-6)}{2(X+3)}$$

$$G_{13} = \frac{X(X-3+j3.873)(\,)}{6(X+4)}$$

$$G_{14} = \frac{X(X-5.438)(X-1.281+j4.520)(\,)}{24(X+5)}$$

$$G_{20} = \frac{X(X+2)}{(X+1+j1)(\,)}$$

$$G_{21} = \frac{X(X+6)}{(X+2+j1.414)(\,)}$$

$$G_{22} = \frac{12X}{(X+3+j1.732)(\,)}$$

$$G_{23} = \frac{X(X-3+j7.141)(\,)}{(X+4+j2)(\,)}$$

$$G_{24} = \frac{X(X-9.648)(X-1.176+j5.994)(\,)}{(X+5+j2.236)(\,)}$$

$$G_{30} = \frac{X(X+1.5+j1.936)()}{(X+1.596)(X+.702+j1.807)()}$$

$$G_{31} = \frac{X(X+3+j3.873)()}{(X+2.626)(X+1.687+j2.509)()}$$

$$G_{32} = \frac{X(X+3+j7.141)()}{(X+3.638)(X+2.681+j3.050)()}$$

$$G_{33} = \frac{2X(X+j7.746)()}{(X+4.644)(X+3.678+j3.509)()}$$

$$G_{34} = \frac{X(X-19.157)(X-.422+j6.080)()}{(X+5.648)(X+4.676+j3.913)()}$$

$$G_{40} = \frac{X(X+2.785)(X+.607+j2.872)()}{(X+1.729+j.889)(X+.270+j2.505)()()}$$

$$G_{41} = \frac{X(X+5.438)(X+1.281+j4.520)()}{(X+2.764+j1.162)(X+1.236+j3.437)()()}$$

$$G_{42} = \frac{X(X+9.648)(X+1.176+j5.994)()}{(X+3.779+j1.380)(X+2.221+j4.160)()()}$$

$$G_{43} = \frac{X(X+19.175)(X+.422+j6.608)()}{(X+4.787+j1.567)(X+3.213+j4.773)()()}$$

$$G_{44} = \frac{40X(X+j6.481)()}{(X+5.792+j1.734)(X+4.208+j5.315)()()}$$

$$G_{50} = \frac{X(X-.184+j3.440)(X+2.684+j1.704)()()}{(X+2.181)(X-.240+j3.128)(X+1.649+j1.694)()()}$$

$$G_{51} = \frac{X(X+.314+j4.844)(X+4.686+j2.932)()()}{(X+3.237)(X+.703+j4.260)(X+2.678+j2.181)()()}$$

$$G_{52} = \frac{X(X+.409+j5.797)(X+7.091+j4.934)()()}{(X+4.262)(X+1.677+j5.138)(X+3.690+j2.571)()()}$$

$$G_{53} = \frac{X(X+.224+j6.242)(X+9.776+j8.755)()()}{(X+5.277)(X+2.665+j5.884)(X+4.697+j2.909)()()}$$

$$G_{54} = \frac{X(X+.058+j6.327)(X+9.942+j16.698)()()}{(X+6.288)(X+3.656+j6.544)(X+5.701+j3.210)()()}$$

$$G_{55} = \frac{2X(X+j6.306)(X+j19.500)()()}{(X+7.293)(X+4.649+j7.142)(X+6.704+j3.485)()()}$$

$$G_{60} = \frac{X(X+3.553)(X-.874+j3.888)(X+2.097+j2.892)()()}{(X+2.362+j.838)(X-.803+j3.698)(X+1.442+j2.434)()()()}$$

$$G_{61} = \frac{X(X+5.921)(X-.336+j5.174)(X+3.376+j4.501)()()}{(X+3.424+j1.047)(X+.117+j5.006)(X+2.458+j3.102)()()()}$$

$$G_{62} = \frac{X(X+8.977)(X-.025+j5.887)(X+4.536+j6.649)()()}{(X+4.454+j1.217)(X+1.082+j6.023)(X+3.464+j3.639)()()()}$$

$$G_{63} = \frac{X(X+.048+j6.186)(X+13.336)(X+5.284+j9.517)()()}{(X+5.471+j1.366)(X+2.061+j6.887)(X+4.467+j4.104)()()()}$$

$$G_{64} = \frac{X(X+20.598)(X+.026+j6.273)(X+4.675+j12.831)()()}{(X+6.482+j1.499)(X+3.048+j7.652)(X+5.469+j4.520)()()()}$$

$$G_{65} = \frac{X(X+37.825)(X+.006+j6.287)(X+2.082+j14.770)()()}{(X+7.491+j1.622)(X+4.039+j8.346)(X+6.470+j4.900)()()()}$$

$$G_{66} = \frac{84X(X+j6.285)(X+j14.160)()()}{(X+8.497+j1.735)(X+5.032+j8.985)(X+7.471+j5.252)()()()}$$

$$G_{70} = \frac{X(X+3.605+j1.577)(X+1.400+j3.700)(X-1.504+j4.306)()()()}{(X+2.759)(X-1.406+j4.225)(X+1.147+j3.124)(X+2.380+j1.629)()()()}$$

$$G_{71} = \frac{X(X+5.684+j2.443)(X-.830+j5.587)(X+2.146+j5.330)()()()}{(X+3.839)(X-.509+j5.695)(X+2.148+j3.950)(X+3.441+j2.012)()()()}$$

$$G_{72} = \frac{X(X-.245+j6.132)(X+8.142+j3.631)(X+2.604+j7.338)()()()}{(X+4.878)(X+.445+j6.837)(X+4.470+j2.325)(X+3.146+j4.616)()()()}$$

$$G_{73} = \frac{X(X-.032+j6.240)(X+2.862+j9.614)(X+11.170+j5.438)()()()}{(X+5.901)(X+1.418+j7.809)(X+5.486+j2.598)(X+4.144+j5.194)()()()}$$

$$G_{74} = \frac{X(X+.012+j6.287)(X+2.500+j11.596)(X+14.988+j8.625)()()()}{(X+5.668)(X+2.206+j8.732)(X+4.928+j4.726)(X+7.532+j4.567)()()()}$$

$$G_{75} = \frac{X(X+.001+j6.282)(X+1.519+j12.934)(X+19.479+j14.730)()()()}{(X+7.928+j3.071)(X+7.506)(X+3.388+j9.450)(X+6.142+j6.184)()()()}$$

$$G_{76} = \frac{X(X+.0004+j6.283)(X+.508+j13.194)(X+20.492+j28.924)()()()}{(X+8.937)(X+8.512+j3.281)(X+4.379+j10.170)(X+7.141+j6.623)()()()}$$

$$G_{77} = \frac{X(X+j6.283)(X+j12.958)(X+j36.119)()()()}{(X+9.943)(X+9.516+j3.478)(X+5.371+j10.841)(X+8.140+j7.034)()()()}$$

APPENDIX 2(A)

```
..JOB0608F,  SEILER,  R.W.
      PROGRAM FACTOR
C        THIS PROGRAM FACTORS N POLYNOMIALS, WITH REAL COEFFICIENTS,
C     OF ORDER 30 OR LESS.  THE CALLING ARGUMENTS ARE
C
C        CARD 0 -   N - RJ COLUMNS 1-5
C     AT LEAST 2 CARDS ARE REQUIRED FOR EACH POLYNOMIAL
C        CARD 1 - ORDER OF POLYNOMIAL - RJ COLUMNS 1-5
C                 ALFANUMERIC IDENTIFIER   COLUMNS 11-50
C        CARD 2 - COEFFICIENTS IN ASCENDING ORDER - 7F10.5 FORMAT
C
      DIMENSION FR(30),FI(30),RR(30),RI(30),IA(5)
      READ 901,N
      DO 900 I=1,N
      READ 904,M,(IA(J),J=1,5)
      MN=M+1
      READ 902,(FR(J),J=1,MN)
      PRINT 907
      PRINT904,M,(IA(J),J=1,5)
      PRINT 906,(FR(J),J=1,MN)
      PRINT 905
      DO 5  J=1,MN
    5 FI(J)=0.0
      CALL POLYRT(FR,FI,M,RR,RI,1.E-6)
      DO 10 J=1,M
   10 PRINT 903,RR(J),RI(J)
  900 CONTINUE
  901 FORMAT(I5)
  902 FORMAT (1X,F9.5,6F10.5 )
  903 FORMAT(10X,F10.5,2X,3H+J ,F10.5)
  904 FORMAT (I5,5X,5A8)
  905 FORMAT(32HIMAGINARY COEFFICIENTS ARE ZERO. ,/)
```

```
906 FORMAT (/,17HREAL COEFFICIENTS   ,/,9(1X,F10.1),/)
907 FORMAT (///)
999 END
    END
```

APPENDIX 2(B)

```
..JOB0608F, SEILER, R.W.
      PROGRAM SPECT
C        THIS PROGRAM CALCULATES AND PLOTS THE SPECTRAL WEIGHTING
C     FUNCTION OF I TRANSFER FUNCTIONS, CHARACTERIZED BY AN N-TH
C     ORDER POLYNOMIAL DIVIDED BY AN M-TH ORDER POLYNOMIAL.  M AND
C     N MUST BE 12 OR LESS.  THE CALLING ARGUMENTS ARE
C
C        CARD 0 -   I - RJ COLUMNS 1-5
C     5 CARDS ARE REQUIRED FOR EACH TRANSFER FUNCTION
C        CARD 1 - ALFANUMERIC IDENTIFIER   COLUMNS 1 8
C                 NUMERATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70  - F10.3 FORMAT
C        CARD 2 - CONTINUATION OF CARD 1    COLUMNS  1-70
C        CARD 3 - DENOMINATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70   - F10.3 FORMAT
C        CARD 4 - CONTINUATION OF CARD 3    COLUMNS  1-70
C        CARD 5 - PAGE NUMBER - COLUMNS 3-10
C
      DIMENSION HSQ(500),GSQ(500),REAL(5  ),ANSW(500),X(500),XX(500),
     1IT(13)
      READ 901,N
      DO 900 M=1,N
      READ 902,IT(5),FN,EN,DN,CN,BN,AN,GN,HN,ON,PN,QN,RN,SN
      READ 902,IT(13),FD,ED,DD,CD,BD,AD,GD,HD,OD,PD,QD,RD,SD
      READ 905,IT(9),IT(11)
      WT=.1
      DO 100 I=1,500
      RT=MODF(WT,6.2831853072)
      Y=SINF(RT)
      Z=COSF(RT)
      HSQ(I)=((1.-Z)**2)+(Y**2)
      TOPR=FN-(DN*(WT**2))+(BN*(WT**4))
```

```
      TOPR=TOPR-(GN*(WT**6))+(ON*(WT**8))-(QN*(WT**10))+(SN*(WT**12))
      TOPI=(EN*(WT))-(CN*(WT**3))+(AN*(WT**5))
      TOPI=TOPI-(HN*(WT**7))+(PN*(WT**9))-(RN*(WT**11))
      BOTR=FD-(DD*(WT**2))+(BD*(WT**4))
      BOTR=BOTR-(GD*(WT**6))+(OD*(WT**8))-(QD*(WT**10))+(SD*(WT**12))
      BOTI=(ED*(WT))-(CD*(WT**3))+(AD*(WT**5))
      BOTI=BOTI-(HD*(WT**7))+(PD*(WT**9))-(RD*(WT**11))
      GSQ(I)=((TOPR**2)+(TOPI**2))/((BOTR**2)+(BOTI**2))
      CALL ATAN (TOPR,TOPI,TOPANG)
      CALL ATAN (BOTR,BOTI,BOTANG)
      PHI=TOPANG-BOTANG
      GJWR=(SQRTF(GSQ(I)))*COSF(PHI)
      GJWI=(SQRTF(GSQ(I)))*SINF(PHI)
      REAL(I)=(GJWR*(1.-Z))+(GJWI*Y)
      ANSW(I)=HSQ(I)+GSQ(I)-(2.*REAL(I))
      X(I)=WT
      XX(I)=LOG10F(WT)
      IF (I-100) 85,85,90
   85 WT=WT+.009
      GO TO 100
   90 IF (I-300) 95,95,99
   95 WT=WT+.045
      GO TO 100
   99 WT=WT+.25
  100 CONTINUE
      PRINT 903,IT(5)
      DO 200 J=1,50
      I=1 (10*(J-1))
      PRINT 904,X(I),HSQ(I),GSQ(I),REAL(I),ANSW(I)
  200 CONTINUE
      LAB=4H
      IT(1)=8H   SPECT
      IT(2)=8HRAL WEIG
      IT(3)=8HHTING FU
      IT(4)=8HNCTION -
      IT(6)=8H
```

```
      IT(7)=8H
      IT(8)=8H  PAGE
      IT(10)=8H -APPEND
      IT(11)=8HIX  3
      IT(12)=8H
      CALL DRAW(500,XX,ANSW,0,0,LAB,IT,0,0,0,0,0,0,6,7,0,LAST)
  900 CONTINUE
  901 FORMAT(I5)
  902 FORMAT (A8,2X,6F10.3,/,7F10.3)
  903 FORMAT (1H1,15X,4(6X,9HAMPLITUDE),/,13X,2HWT,7X,8HIDEAL SQ,5X,
     1A8,3H SQ,4X,11HG*(JW)I(JW),5X,6HANSWER,/)
  904 FORMAT (7(5X,F10.3))
  905 FORMAT(2(2X,A8))
      END
      SUBROUTINE ATAN (X,Y,AAG)
      W=3.1415926536
      IF (X) 30,10,20
   10 IF (Y) 12,14,16
   12 AAG=W*3./2.
      RETURN
   14 AAG= 0.
      RETURN
   16 AAG=W/2.
      RETURN
   20 Z= ATANF (ABSF (Y/X) )
      IF (Y) 22,24,24
   22 AAG= (2.*W) - Z
      RETURN
   24 AAG= Z
      RETURN
   30 Z= ATANF (ABSF (Y/X) )
      IF (Y) 32,32,34
   32 AAG=Z+W
      RETURN
   34 AAG=W-Z
      END
```

APPENDIX 2(C)

```
..JOB0608F, SEILER, R.W.
      PROGRAM MEANSQER
C        THIS PROGRAM CALCULATES THE MEAN SQUARE ERROR FOR I TRANSFER
C     FUNCTIONS, CHARACTERIZED BY AN N-TH ORDER POLYNOMIAL DIVIDED
C     BY AN M-TH ORDER POLYNOMIAL.  THE INPUT TO THE FILTER IS A
C     UNIT STEP. THE CALLING ARGUMENTS ARE
C
C        CARD 0 -   I - RJ COLUMNS 1-5
C     4 CARDS ARE REQUIRED FOR EACH TRANSFER FUNCTION
C        CARD 1 - ALFANUMERIC IDENTIFIER   COLUMNS 1-8
C                 NUMERATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70  - F10.3 FORMAT
C        CARD 2 - CONTINUATION OF CARD 1    COLUMNS  1-70
C        CARD 3 - DENOMINATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70   - F10.3 FORMAT
C        CARD 4 - CONTINUATION OF CARD 3    COLUMNS  1-70
C
      COMMON FN,EN,DN,CN,BN,AN,GN,HN,ON,PN,QN,RN,SN,FD,ED,DD,CD,BD,AD,
     1GD,HD,OD,PD,QD,RD,SD
      READ 901,N
      DO 900 M=1,N
      PRINT 902
      READ 903,IT,FN,EN,DN,CN,BN,AN,GN,HN,ON,PN,QN,RN,SN
      READ 903,JT,FD,ED,DD,CD,BD,AD,GD,HD,OD,PD,QD,RD,SD
      CALL SIMPS (.01,1.,FUNC,100,1.E-4,1000,W)
      CALL SIMPS (1.,10.,FUNC,100,1.E-4,1000,X)
      CALL SIMPS (10.,100.,FUNC,100,1.E-4,1000,Y)
      CALL SIMPS (100.,1000.,FUNC,100,1.E-4,1000,Z)
      AREA=2.*(W+X+Y+Z)
      PRINT 904,IT
      PRINT 905,W,X,Y,Z,AREA
  900 CONTINUE
```

```
901 FORMAT (I5)
902 FORMAT (1H1)
903 FORMAT (A8,2X,6F10.3,/,7F10.3)
904 FORMAT (/////,10X,A8)
905 FORMAT (//,10X,46HPHI(WT) (FROM WT=   0.   TO    0.01)= APPROX 0,
   1         /,10X,37HPHI(WT) (FROM WT=   0.01 TO    1.0 )=,F10.4,
   2         /,10X,37HPHI(WT) (FROM WT=   1.0  TO   10.0 )=,F10.4,
   3         /,10X,37HPHI(WT) (FROM WT=  10.0  TO  100.0 )=,F10.4,
   4         /,10X,37HPHI(WT) (FROM WT= 100.0  TO 1000.0 )=,F10.4,
   5         /,10X,47HPHI(WT) (FROM WT=1000.0  TO INF.   )= APPROX 0 ,
   6        //,10X,35H  MEAN - SQUARED ERROR              =, F10.4)
    END
    SUBROUTINE SIMPS(A,B,FUNC,N,E,L,ANS)                                    00000000
    SUM1=0.                                                                 0000016
    SUM2=0.                                                                 0000017
    PREV=0.                                                                 0000018
    N=XMAXOF(N+XMODF(N+1,2),3)                                              0000019
    CALL FOFX (A,ANS1)
    CALL FOFX (B,ANS2)
    SFL=ANS1+ANS2
    DEL=(B-A)/FLOATF(N-1)                                                   0000022
    IF (L) 5,10,15                                                          0000023
5   IX=1                                                                    0000024
    GO TO 20                                                                0000025
10  IX=2                                                                    0000026
    GO TO 20                                                                0000027
15  IX=3                                                                    0000028
20  K=N 2                                                                   0000029
    DO 22 I=2,K,2                                                           0000030
    X=A DEL*FLOATF(I)                                                       0000031
    CALL FOFX (X,ANS3)
 22 SUM1=SUM1+ANS3
23  DO 25 I=1,K,2                                                           0000033
    X=A DEL*FLOATF(I)                                                       0000034
    CALL FOFX (X,ANS3)
 25 SUM2=SUM2+ANS3
```

APPENDIX 2(C)

```
..JOB0608F, SEILER, R.W.
      PROGRAM MEANSQER
C        THIS PROGRAM CALCULATES THE MEAN SQUARE ERROR FOR I TRANSFER
C     FUNCTIONS, CHARACTERIZED BY AN N-TH ORDER POLYNOMIAL DIVIDED
C     BY AN M-TH ORDER POLYNOMIAL.  THE INPUT TO THE FILTER IS A
C     UNIT STEP. THE CALLING ARGUMENTS ARE
C
C        CARD 0 -   I - RJ COLUMNS 1-5
C     4 CARDS ARE REQUIRED FOR EACH TRANSFER FUNCTION
C        CARD 1 - ALFANUMERIC IDENTIFIER   COLUMNS 1-8
C                 NUMERATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70  - F10.3 FORMAT
C        CARD 2 - CONTINUATION OF CARD 1    COLUMNS  1-70
C        CARD 3 - DENOMINATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70   - F10.3 FORMAT
C        CARD 4 - CONTINUATION OF CARD 3    COLUMNS  1-70
C
      COMMON FN,EN,DN,CN,BN,AN,GN,HN,ON,PN,QN,RN,SN,FD,ED,DD,CD,BD,AD,
     1GD,HD,OD,PD,QD,RD,SD
      READ 901,N
      DO 900 M=1,N
      PRINT 902
      READ 903,IT,FN,EN,DN,CN,BN,AN,GN,HN,ON,PN,QN,RN,SN
      READ 903,JT,FD,ED,DD,CD,BD,AD,GD,HD,OD,PD,QD,RD,SD
      CALL SIMPS (.01,1.,FUNC,100,1.E-4,1000,W)
      CALL SIMPS (1.,10.,FUNC,100,1.E-4,1000,X)
      CALL SIMPS (10.,100.,FUNC,100,1.E-4,1000,Y)
      CALL SIMPS (100.,1000.,FUNC,100,1.E-4,1000,Z)
      AREA=2.*(W+X+Y+Z)
      PRINT 904,IT
      PRINT 905,W,X,Y,Z,AREA
  900 CONTINUE
```

```
      ANS=(SFL+4.*SUM2+2.*SUM1)*DEL/3.
      GO TO (50,40,30),IX
30    IF (L-N) 50,50,40
40    IF (E*ABSF(ANS)-ABSF(ANS-PREV)) 45,50,50
45    PREV=ANS
      N=2 *N-1
      K=N 2
      SUM1=SUM1+SUM2
      SUM2=0.
      DEL=DEL/2.
      GO TO 23
50    RETURN
      END
      SUBROUTINE FOFX (WT,Q)
      COMMON FN,EN,DN,CN,BN,AN,GN,HN,ON,PN,QN,RN,SN,FD,ED,DD,CD,BD,AD,
     1GD,HD,OD,PD,QD,RD,SD
      RT=MODF(WT,6.2831853072)
      R=SINF(RT)
      S=COSF(RT)
      HSQ=((1.-S)**2)+(R**2)
      TOPR=FN-(DN*(WT**2))+(BN*(WT**4))
      TOPR=TOPR-(GN*(WT**6))+(ON*(WT**8))-(QN*(WT**10))+(SN*(WT**12))
      TOPI=(EN*(WT))-(CN*(WT**3))+(AN*(WT**5))
      TOPI=TOPI-(HN*(WT**7))+(PN*(WT**9))-(RN*(WT**11))
      BOTR=FD-(DD*(WT**2))+(BD*(WT**4))
      BOTR=BOTR-(GD*(WT**6))+(OD*(WT**8))-(QD*(WT**10))+(SD*(WT**12))
      BOTI=(ED*(WT))-(CD*(WT**3))+(AD*(WT**5))
      BOTI=BOTI-(HD*(WT**7))+(PD*(WT**9))-(RD*(WT**11))
      GSQ=((TOPR**2)+(TOPI**2))/((BOTR**2)+(BOTI**2))
      CALL ATAN (TOPR,TOPI,TOPANG)
      CALL ATAN (BOTR,BOTI,BOTANG)
      PHI=TOPANG-BOTANG
      GJWR=(SQRTF(GSQ))*COSF(PHI)
      GJWI=(SQRTF(GSQ))*SINF(PHI)
      REAL=(GJWR*(1.-S))+(GJWI*R)
      ANSW=HSQ+GSQ-(2.*REAL)
```

```
      Q=ANSW/(2.*3.14159*(WT**2))
      RETURN
      END
      SUBROUTINE ATAN (X,Y,AAG)
      W=3.1415926536
      IF (X) 30,10,20
   10 IF (Y) 12,14,16
   12 AAG=W*3./2.
      RETURN
   14 AAG= 0.
      RETURN
   16 AAG=W/2.
      RETURN
   20 Z= ATANF (ABSF (Y/X) )
      IF (Y) 22,24,24
   22 AAG= (2.*W) - Z
      RETURN
   24 AAG= Z
      RETURN
   30 Z= ATANF (ABSF (Y/X) )
      IF (Y) 32,32,34
   32 AAG=Z+W
      RETURN
   34 AAG=W-Z
      END
      END
```

APPENDIX 2(D)

```
..JOB0608F, SEILER, R.W.
      PROGRAM INSQERR
C        THIS PROGRAM COMPUTES THE INTEGRAL SQUARE ERROR FOR N TRANSFER
C     FUNCTIONS OF THE FORM
C
C            BT      -DT              -HT
C        A*E   + C*E   COS(ET+F) + G*E   COS(IT+J)
C
C     THE CALLING ARGUMENTS ARE
C
C        CARD 0 -   N - RJ COLUMNS 1-5
C     3 CARDS ARE REQUIRED FOR EACH TRANSFER FUNCTION
C        CARD 1 - ALFANUMERIC IDENTIFIER   COLUMNS 1-10
C                 A AND B - 2F10.3 FORMAT
C        CARD 2 - C,D,E, AND F - 4F10.3 FORMAT
C        CARD 3 - G,H,I, AND J - 4F10.3 FORMAT
      COMMON A,B,C,D,E,THE,F,G,H,PSI,M
      READ 901,N
      PRINT 904
      DO 900 I=1,N
      READ 902,J,K,A,B,C,D,E,THE,F,G,H,PSI
      IF(D) 2,5,2
    2 IF(B)3,10,3
    3 IF(B-D)5,5,10
    5 XL=20./B
      GO TO 15
   10 XL=20./D
   15 M=0
      IF (XL-1.) 17,20,20
   17 XL=1.
   20 DEL=1.E-4
      CALL NEWTN (-1,0.,1.,Y,DEL,100)
```

```
      M=1
      REL=1.E-4
      CALL NEWTN (-1,0.,XL,X,REL,200)
      ERR=1.-2.*Y+X
      PRINT 903,J,K,Y,X,ERR,DEL,REL,XL
  900 CONTINUE
  901 FORMAT(I5)
  902 FORMAT (2A5,2F10.3,/,4F10.3,/,4F10.3)
  903 FORMAT (2A5,3(5X,F10.4),2(5X,E10.3),5X,F10.3)
  904 FORMAT (1H1,15X,8HINT G(T),5X,11HINT G**2(T),5X,10HINT SQ ERR,5X,
     110HACCUR G(T),5X,10HACCUR G**2,5X,10H UPPER LIM )
  905 FORMAT(5X,11F10.3)
      END
      SUBROUTINE FOFX(T,FX)
      COMMON A,B,C,D,E,THE,F,G,H,PSI,M
      AA= .
      AB= .
      AC= .
      AD= .
      AE= .
      AF= .
      IF(M) 1000,1000,1010
 1000 AA=A*EXPF(-B*T)
      AB=C*EXPF(-D*T)*COSF(E*T+THE)
      AC=F*EXPF(-G*T)*COSF(H*T+PSI)
      FX=AA+AB+AC
      RETURN
 1010 AA=(A**2)*EXPF(-2.*B*T)
      AB=(C**2)*(EXPF(-2.*D*T))*((COSF(E*T+THE))**2)
      AC=(F**2)*(EXPF(-2.*G*T))*((COSF(H*T+PSI))**2)
      AD=2.*A*C*(EXPF(-(B+D)*T))*(COSF(E*T+THE))
      AE=2.*A*F*(EXPF(-(B+G)*T))*(COSF(H*T+PSI))
      AF=2.*C*F*(EXPF(-(D+G)*T))*(COSF(E*T+THE))*(COSF(H*T+PSI))
      FX=AA+AB+AC+AD+AE+AF
      END
      SUBROUTINE NEWTN(INIT,XO,XL,Y,REL,NP)                                 00000000
```

```
      DIMENSION HH(6),YBAR(10),BYB(10)
      IF(INIT)1,2,2
1     INIT = -INIT
      HH(1) = 19.
      HH(2) = 75.
      HH(3) = 50.
      HH(4) = 50.
      HH(5) = 75.
      HH(6) = 19.
      D = 144.
      NG = 5
      ENG = NG
      COEFF = ENG/(2.*D)
2     Y = 0.
      XLGTH = XL - XO
      IF(XLGTH)201,105,201
201   NPP = NP
      DO 103 K = 1,10
      Y = 0.
      ANP = NG * NP
      H = XLGTH/ANP
      FF = COEFF*H
      X = XO
      CALL FOFX(X,FX)
      AREA = HH(1) * FX
      DO 200 L = 1,NP
      GG = NG * (L-1)
      DO 100 J = 1,NG
      AJ = J
      X = XLGTH * ((GG + AJ)/ ANP) + XO
      CALL FOFX(X,FX)
      AREA = HH(J+1) * FX + AREA
  100 CONTINUE
      Y = Y + AREA
      AREA = HH(NG+1) * FX
200   CONTINUE
```

```
      Y = FF * Y
      YBAR(K) = Y
      IF(K-1)104,104,144
144   BYB(K-1) = ABSF(YBAR(K-1) - Y)
      IF(BYB(K-1) - REL*ABSF(Y))105,105,104
104   NP = 2*NP
103   CONTINUE
      DO 108 L = 1,10
      REL = 2.*REL
      DO 107 K = 2,10
      IF(BYB(K-1) - REL*ABSF(YBAR(K)))106,106,107
107   CONTINUE
108   CONTINUE
      K = 10
106   NP = (2**(K-1)) * NPP
      Y = YBAR(K)
105   RETURN
      END
      END
```

APPENDIX 2(E)

```
..JOB0608F, SEILER, R.W.
      PROGRAM SPECT
C        THIS PROGRAM CALCULATES AND PLOTS THE POWER DENSITY
C     SPECTRUM OF I TRANSFER FUNCTIONS, CHARACTERIZED BY AN N-TH
C     ORDER POLYNOMIAL DIVIDED BY AN M-TH ORDER POLYNOMIAL.  M AND
C     N MUST BE 12 OR LESS.  I MUST BE 7 OR LESS.
C     THE CALLING ARGUMENTS ARE
C
C        CARD 0 -   I - RJ COLUMNS 1-5
C     5 CARDS ARE REQUIRED FOR EACH TRANSFER FUNCTION
C        CARD 1 - ALFANUMERIC IDENTIFIER   SAME FOR ALL I - COLUMNS 1-8
C                 NUMERATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70  - F10.3 FORMAT
C        CARD 2 - CONTINUATION OF CARD 1    COLUMNS  1-70
C        CARD 3 - DENOMINATOR COEFFICIENTS IN ASCENDING ORDER - COLUMNS
C                  11-70   - F10.3 FORMAT
C        CARD 4 - CONTINUATION OF CARD 3    COLUMNS  1-70
C        CARD 5 - FIGURE NUMBER - COLUMNS 13-20
C                 PAGE NUMBER - COLUMNS 3-10
C
      DIMENSION HSQ(500),GSQ(500),REAL(5  ),ANSW(500),X(500),XX(500),
     1IT(13) ,AX(500,7),AXX(500,7)
      READ 901,NNM
      DO 900 NMN=1,NNM
      READ 901,N
      IF (N-7) 2,2,999
    2 DO 200 M=1,N
      READ 902,IT(5),FN,EN,DN,CN,BN,AN,GN,HN,ON,PN,QN,RN,SN
      READ 902,IT(13),FD,ED,DD,CD,BD,AD,GD,HD,OD,PD,QD,RD,SD
      READ 905,IT(9),IT(11)
      WT=.1
      DO 100 I=1,500
```

```
      RT=MODF(WT,6.2831853072)
      Y=SINF(RT)
      Z=COSF(RT)
      HSQ(I)=((1.-Z)**2)+(Y**2)
      TOPR=FN-(DN*(WT**2))+(BN*(WT**4))
      TOPR=TOPR-(GN*(WT**6))+(ON*(WT**8))-(QN*(WT**10))+(SN*(WT**12))
      TOPI=(EN*(WT))-(CN*(WT**3))+(AN*(WT**5))
      TOPI=TOPI-(HN*(WT**7))+(PN*(WT**9))-(RN*(WT**11))
      BOTR=FD-(DD*(WT**2))+(BD*(WT**4))
      BOTR=BOTR-(GD*(WT**6))+(OD*(WT**8))-(QD*(WT**10))+(SD*(WT**12))
      BOTI=(ED*(WT))-(CD*(WT**3))+(AD*(WT**5))
      BOTI=BOTI-(HD*(WT**7))+(PD*(WT**9))-(RD*(WT**11))
      GSQ(I)=((TOPR**2)+(TOPI**2))/((BOTR**2)+(BOTI**2))
      CALL ATAN (TOPR,TOPI,TOPANG)
      CALL ATAN (BOTR,BOTI,BOTANG)
      PHI=TOPANG-BOTANG
      GJWR=(SQRTF(GSQ(I)))*COSF(PHI)
      GJWI=(SQRTF(GSQ(I)))*SINF(PHI)
      REAL(I)=(GJWR*(1.-Z))+(GJWI*Y)
      ANSW(I)=HSQ(I)+GSQ(I)-(2.*REAL(I))
      ANSW(I)=ANSW(I)/(2.*3.14159*(WT**2))
      X(I)=WT
      XX(I)=LOG10F(WT)
      AX(I,M)=ANSW(I)
      AXX(I,M)=XX(I)
      IF (I-100) 85,85,90
   85 WT=WT+.009
      GO TO 100
   90 IF (I-300) 95,95,99
   95 WT=WT+.045
      GO TO 100
   99 WT=WT+.25
  100 CONTINUE
      PRINT 903,IT(5)
      DO 200 J=1,50
      I=1 (10*(J-1))
```

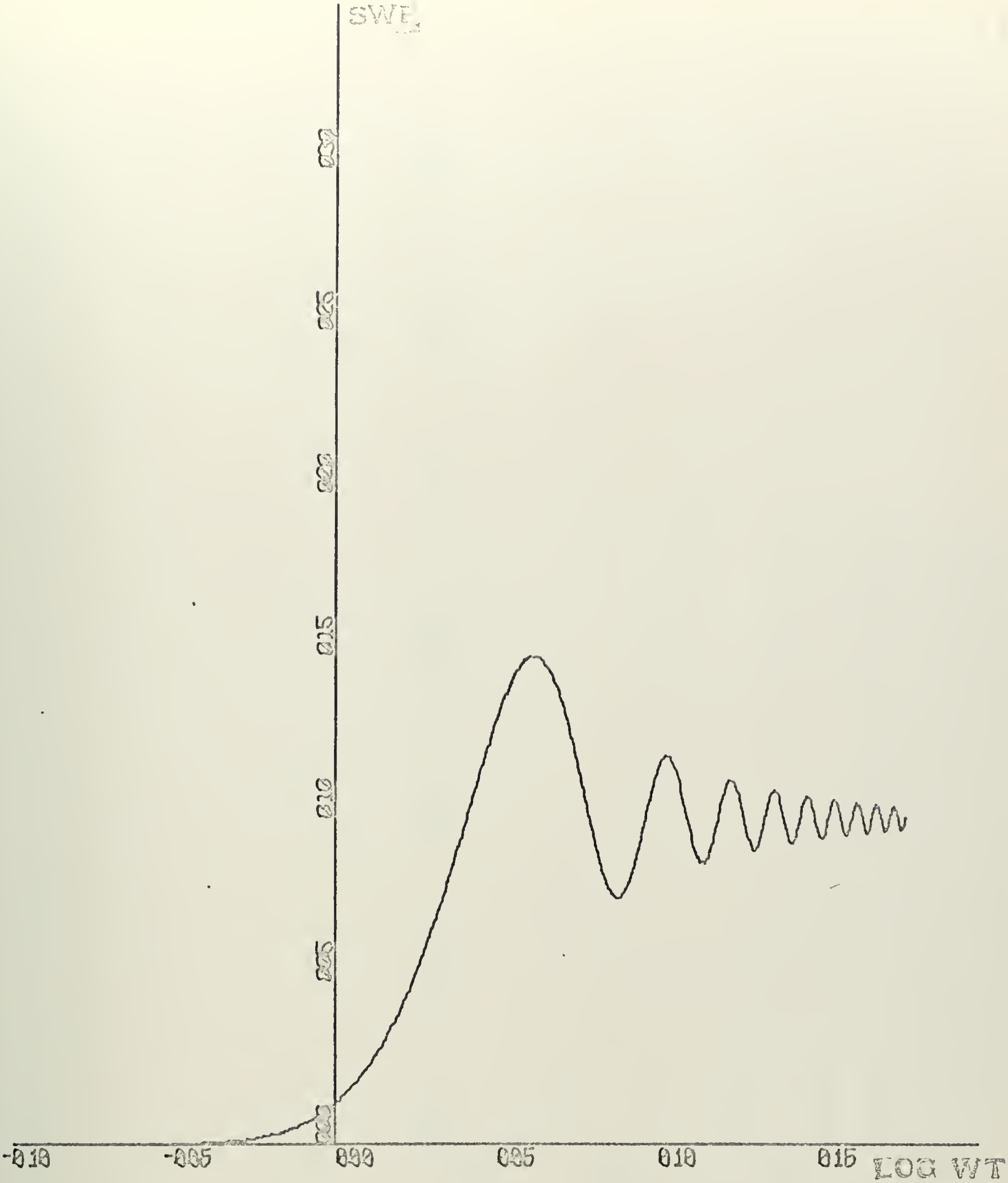
```
      PRINT 904,X(I),HSQ(I),GSQ(I),REAL(I),ANSW(I)
  200 CONTINUE
      LAB=4H
      IT(1)=8H   POWER
      IT(2)=8H DENSITY
      IT(3)=8H SPECTRU
      IT(4)=8HM  -
      IT(6)=8H
      IT(7)=8H
      IT(8)=8H  PAGE
      IT(10)=8H -FIGURE
      IT(12)=8H
      IF (N-1) 999,250,310
  250 DO 300 I=1,500
      X(I)=AX(I,1)
  300 XX(I)=AXX(I,1)
      CALL DRAW (500,XX,X,0,0,LAB,IT,0,0,0,0,0,0,6,7,0,LAST)
      GO TO 999
  310 DO 325 I=1,500
      X(I)=AX(I,1)
  325 XX(I)=AXX(I,1)
      CALL DRAW (500,XX,X,1,0,LAB,IT,0,0,0,0,0,0,6,7,0,LAST)
      NOD=N-1
  340 IF (NOD-1) 999,400,350
  350 DO 375 I=1,500
      X(I)=AX(I,NOD)
  375 XX(I)=AXX(I,NOD)
      CALL DRAW (500,XX,X,2,0,LAB,IT,0,0,0,0,0,0,6,7,0,LAST)
      NOD=NOD-1
      GO TO 340
  400 DO 425 I=1,500
      X(I)=AX(I,N)
  425 XX(I)=AXX(I,N)
      CALL DRAW (500,XX,X,3,0,LAB,IT,0,0,0,0,0,0,6,7,0,LAST)
  900 CONTINUE
  901 FORMAT(I5)
```

```
902 FORMAT (A8,2X,6F10.3,/,7F10.3)
903 FORMAT (1H1,15X,4(6X,9HAMPLITUDE),/,13X,2HWT,7X,8HIDEAL SQ,5X,
   1A8,3H SQ,4X,11HG*(JW)I(JW),5X,6HANSWER,/)
904 FORMAT (7(5X,F10.3))
905 FORMAT(2(2X,A8))
999 END
    SUBROUTINE ATAN (X,Y,AAG)
    W=3.1415926536
    IF (X) 30,10,20
 10 IF (Y) 12,14,16
 12 AAG=W*3./2.
    RETURN
 14 AAG= 0.
    RETURN
 16 AAG=W/2.
    RETURN
 20 Z= ATANF (ABSF (Y/X) )
    IF (Y) 22,24,24
 22 AAG= (2.*W) - Z
    RETURN
 24 AAG= Z
    RETURN
 30 Z= ATANF (ABSF (Y/X) )
    IF (Y) 32,32,34
 32 AAG=Z+W
    RETURN
 34 AAG=W-Z
    END
    END
```

```
      0100                 org    100
0100  2200                 ldc    500
0101  0500
0102  4060                 std    first
0103  4234                 stf    firstl
0104  2200                 ldc
0105  0000                          ---   for a 1 sec delay enter 0613
0106  4061                 std    last           2 sec delay enter 0651
0107  4231                 stf    lastl          3 sec delay enter 0702
0110  2200                 ldc    200            4 sec delay enter 0726
0111  0200
0112  4062                 std    listop
0113  2200                 ldc    150
0114  0150
0115  4063                 std    delay
0116  2061                 lda    last
0117  4214                 stf    out
0120  0601                 adn    1
0121  4210                 stf    outl
0122  7500         start   exf
0123  1401                        1401
0124  7600                 ina
0125  4160                 sti    first
0126  7500                 exf
0127  2401                        2401
0130  7303                 out    3
0131  0000         outl           0
0132  6102                 nzf    2
0133  0000         out            0
0134  0101                 pta
0135  7062                 jpi    listop
0136  0000                        0
0137  0000         firstl         0
0140  0000         lastl          0
0141  0101                 pta
0142  7063                 jpi    delay
0143  0000                        0        enter number of 10 msec delays
0144  0101                 pta               -only required for very long
0145  6523                 nzb    start            -delays
0146  0000                        0
0147  0000         store          0        SUBROUTINE DELAY
0150  0602                 adn    2
0151  4064                 std    templ
0152  2164                 ldi    templ
0153  4304                 stb    store
0154  5464                 aod    templ
0155  2306                 ldb    store
0156  6016                 zjf    endd
0157  2200                 ldc    404
0160  0404
0161  4065                 std    temp2
```

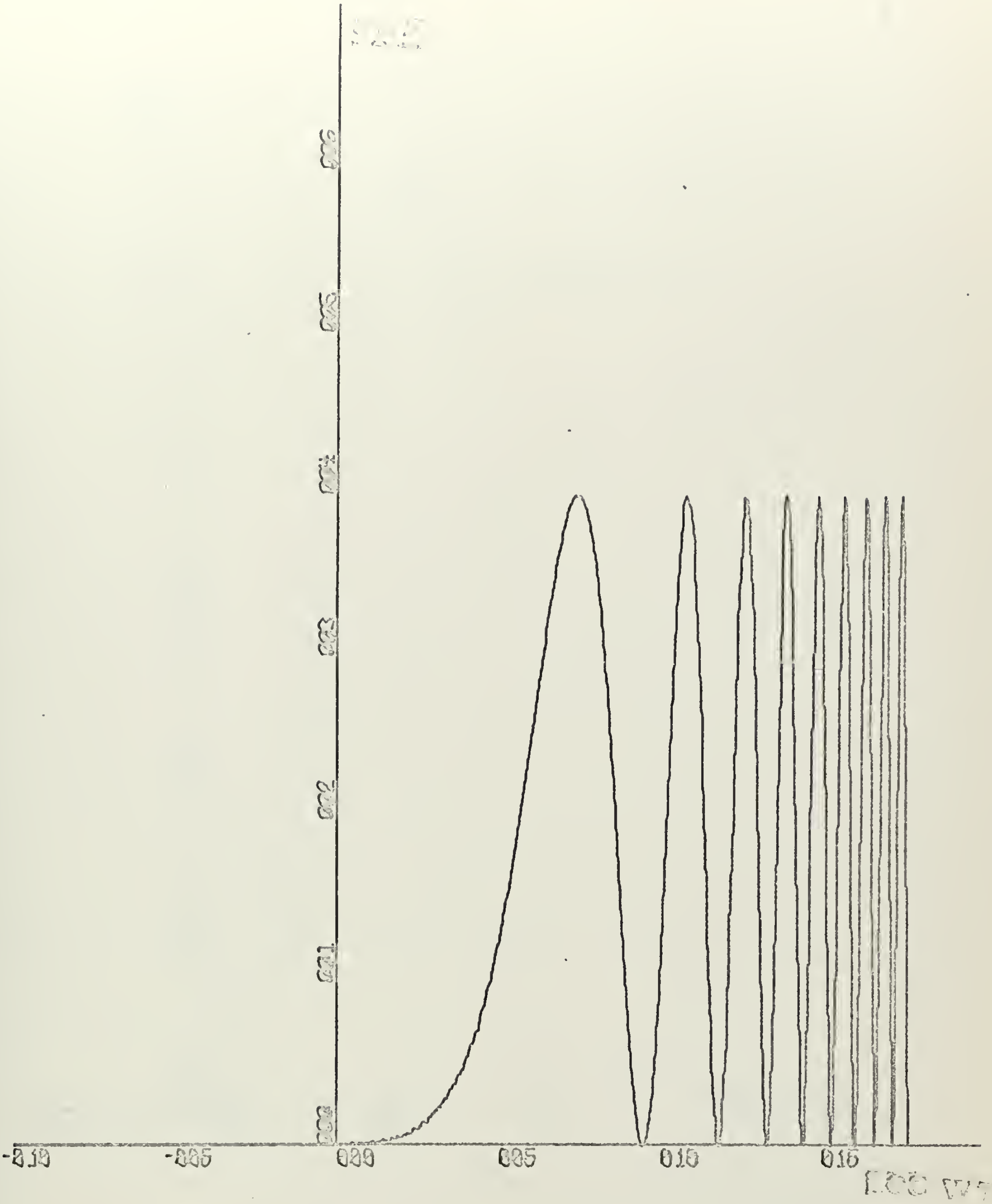
```
0162  0400           ldn    0
0163  4066           std    temp3
0164  0400   loop    ldn    0
0165  4067           std    temp4
0166  5467           aod    temp4
0167  3465           sbd    temp2
0170  6502           nzb    2
0171  5466           aod    temp3
0172  3723           sbb    store
0173  6507           nzb    loop
0174  7064   endd    jpi    templ
      0060   first   equ    60
      0061   last    equ    61
      0062   listop  equ    62
      0063   delay   equ    63
      0064   templ   equ    64
      0065   temp2   equ    65
      0066   temp3   equ    66
      0067   temp4   equ    67
0175  0000                  0
0176  0000                  0
0177  0000                  0
0200  0602                  602     SUBROUTINE LISTOP
0201  4076                  4076
0202  2176                  2176
0203  6304                  6304
0204. 0401                  401
0205  4255                  4255
0206  6103                  6103
0207  0501                  501
0210  4252                  4252
0211  2176                  2176
0212  0201                  201
0213  4250                  4250
0214  5476                  5476
0215  2176                  2176
0216  4240                  4240
0217  5476                  5476
0220  2176                  2176
0221  4240                  4240
0222  5476                  5476
0223  4235                  4235
0224  2236                  2236
0225  6207                  6207
0226  2230                  2230
0227  4076                  4076
0230  2231                  2231
0231  4225                  4225
0232  2076                  2076
0233  4226                  4226
0234  0222                  222
0235  4075                  4075
0236  2175                  2175
```

```
0237  4076                      4076
0240  2077                      2077
0241  4175                      4175
0242  2076                      2076
0243  4077                      4077
0244  2075                      2075
0245  1614                      1614
0246  6004                      6004
0247  2213                      2213
0250  5075                      5075
0251  6513                      6513
0252  2211                      2211
0253  6004                      6004
0254  2077                      2077
0255  4100                      4100
0256  0000                      0
0257  7101                      7101
      0000          end
```

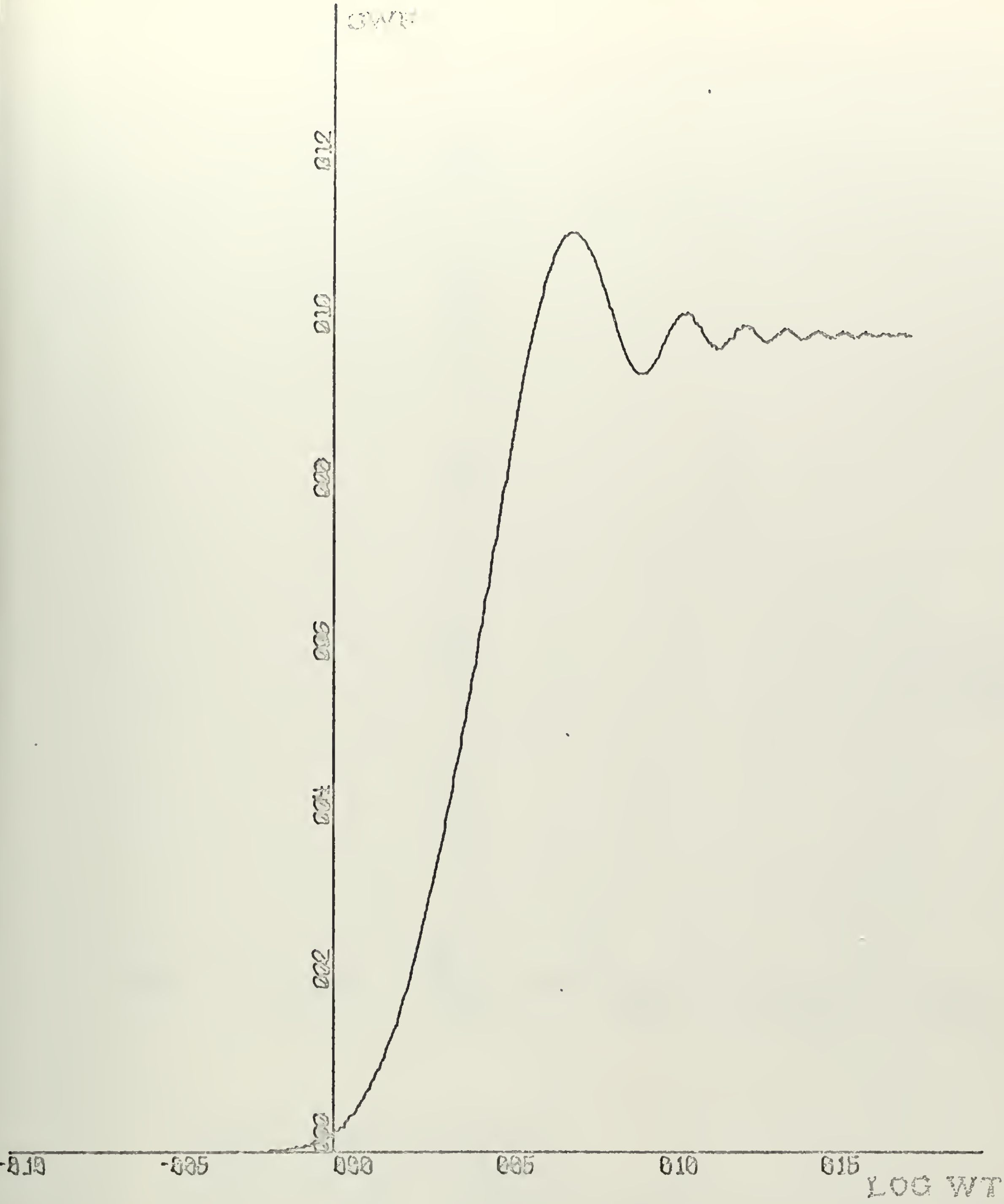


-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(10)
PAGE -APPENDIX 3

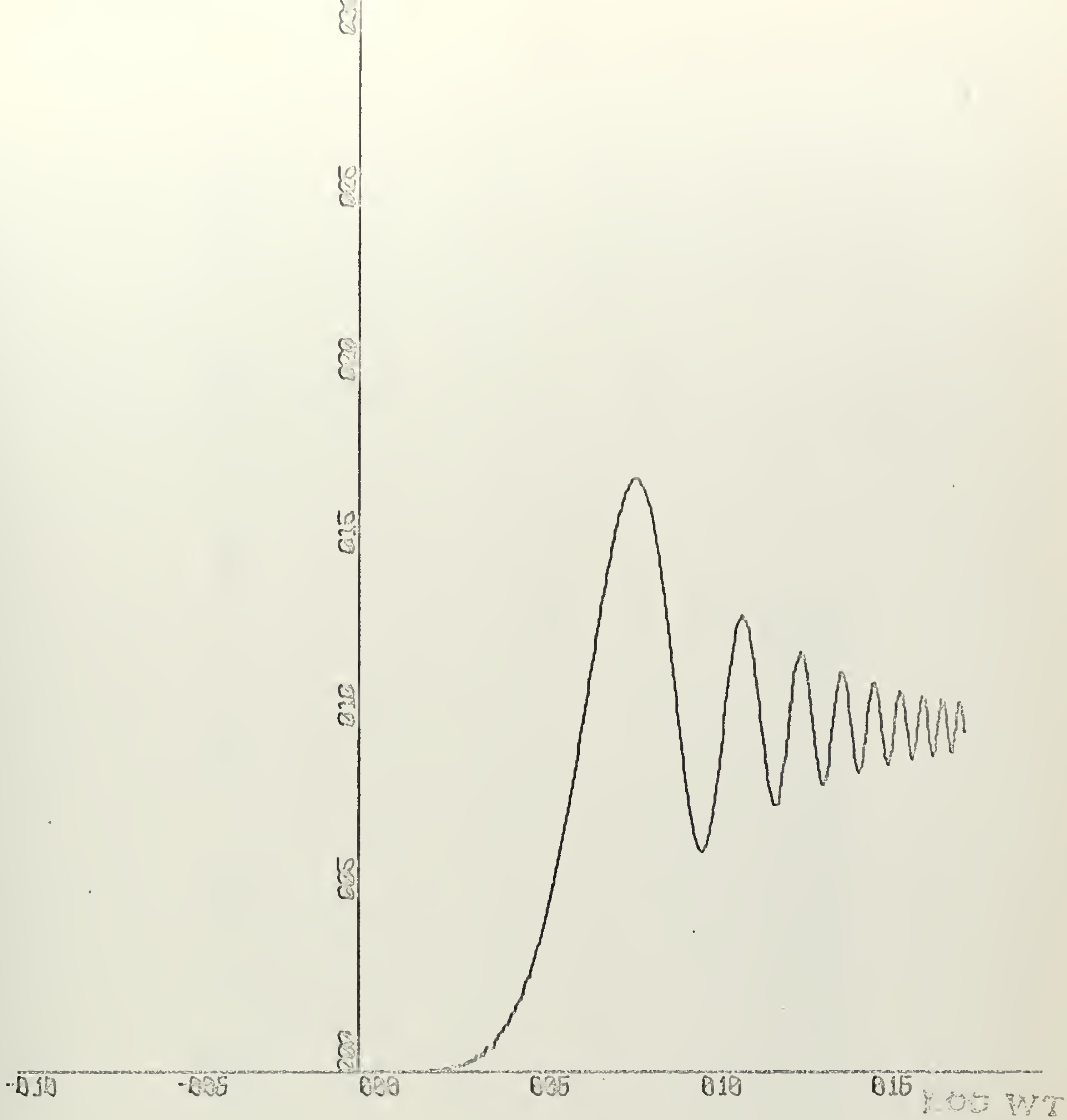


X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E+00 UNITS/INCH.
SPECTRAL WEIGHTING FUNCTION -G(11)
PAGE -APPENDIX 3



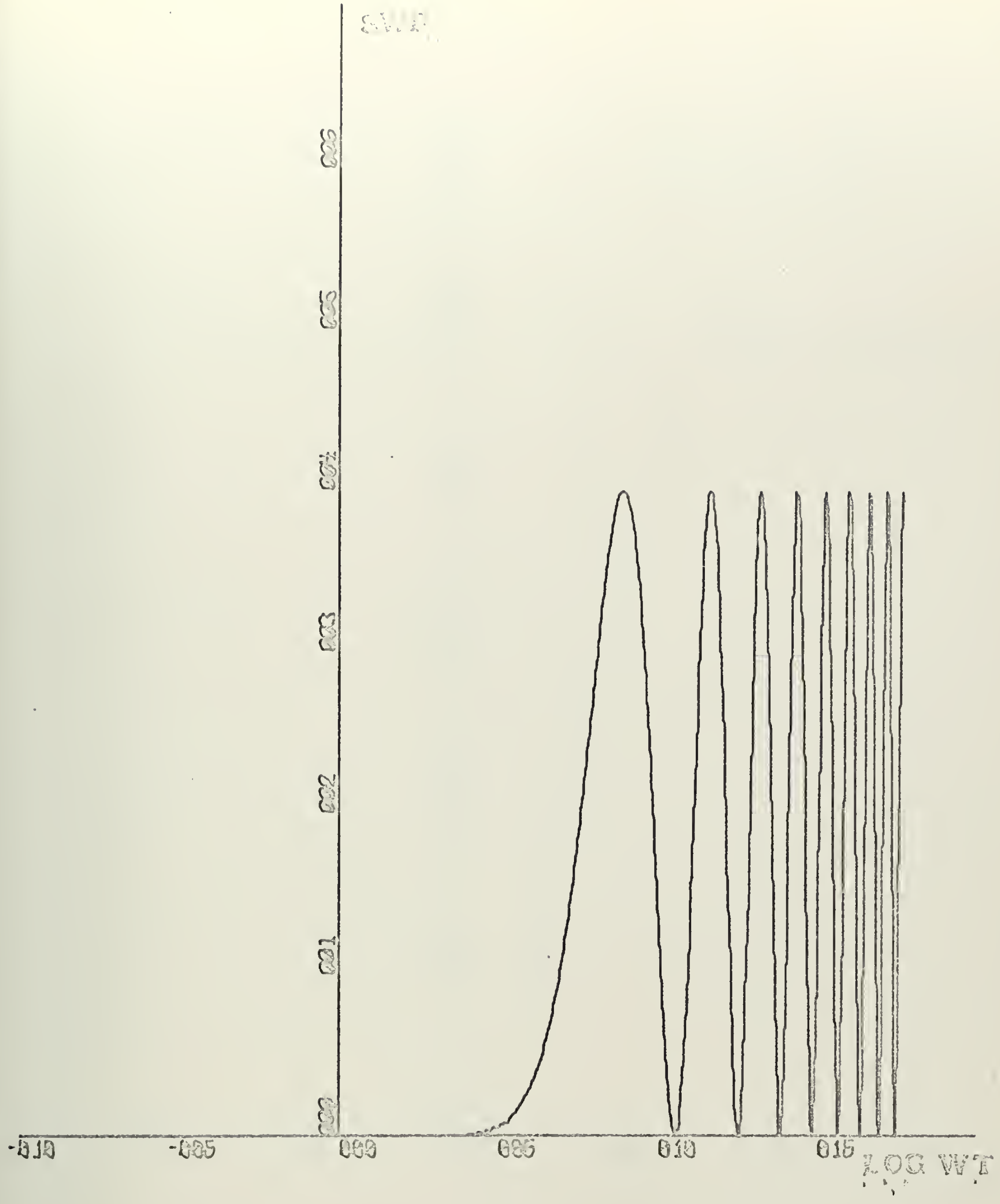
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(20)
PAGE -APPENDIX 3



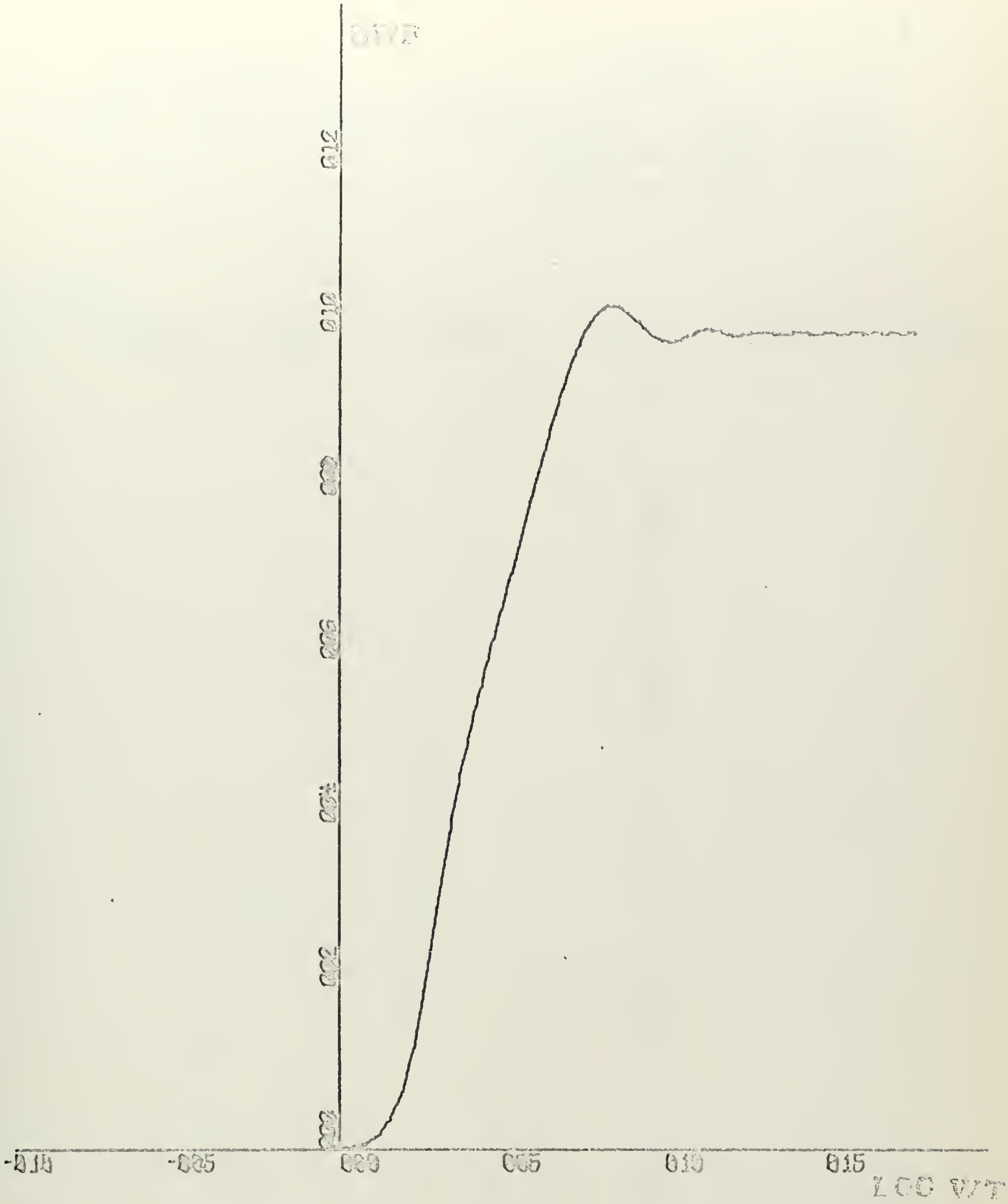
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(21)
PAGE -APPENDIX 3



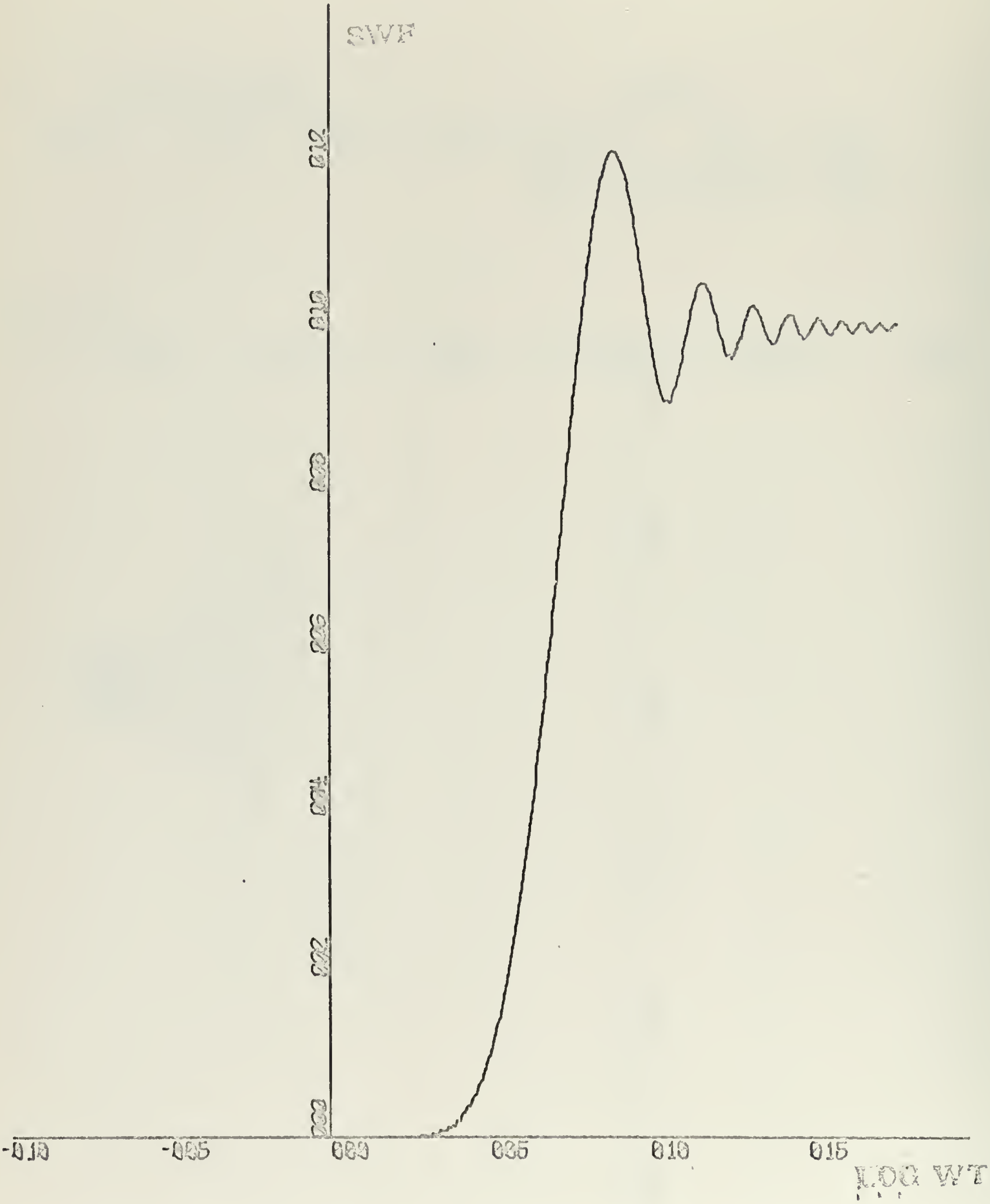
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E+00 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(22)
PAGE -APPENDIX 3



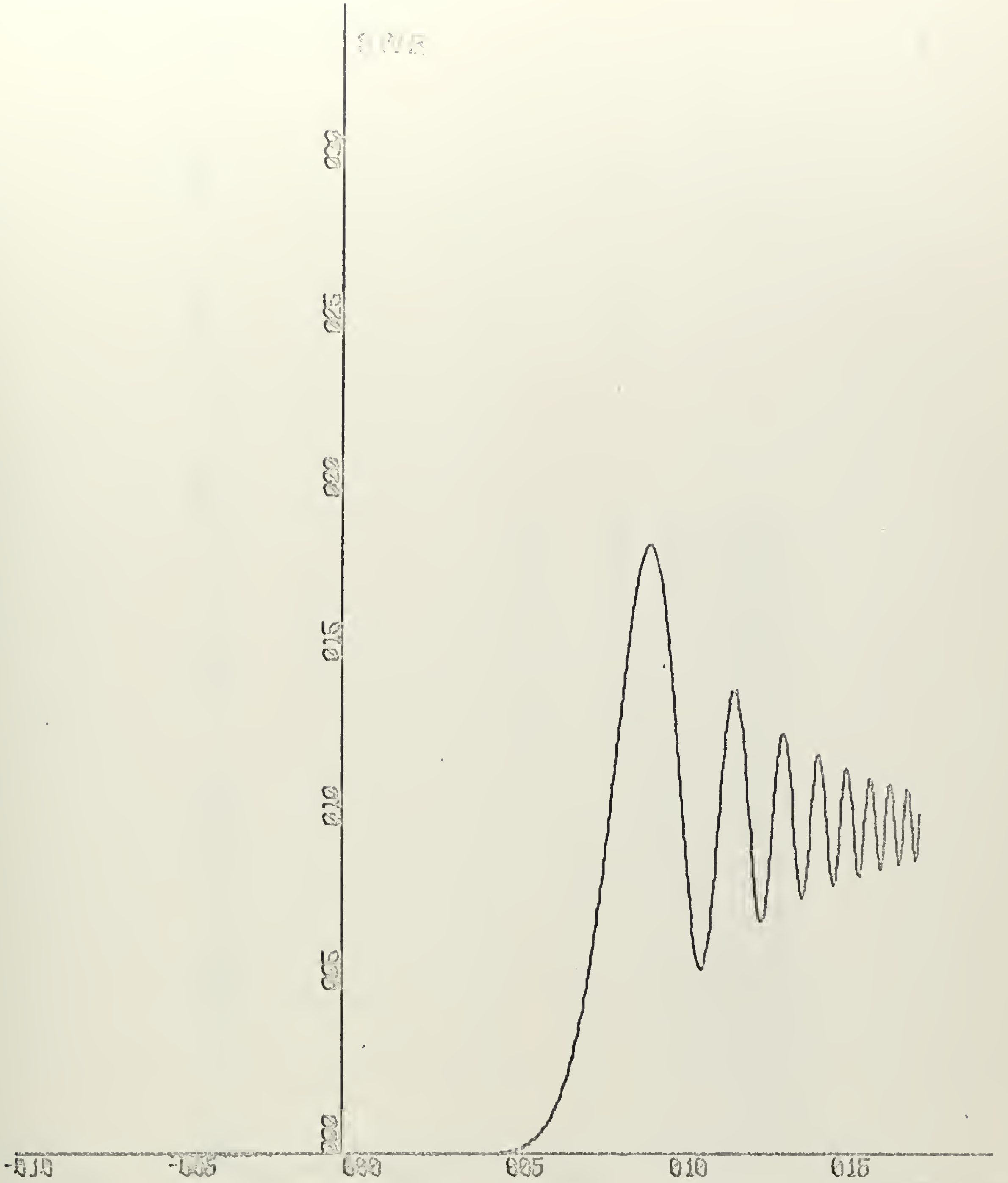
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(30)
PAGE -APPENDIX 3



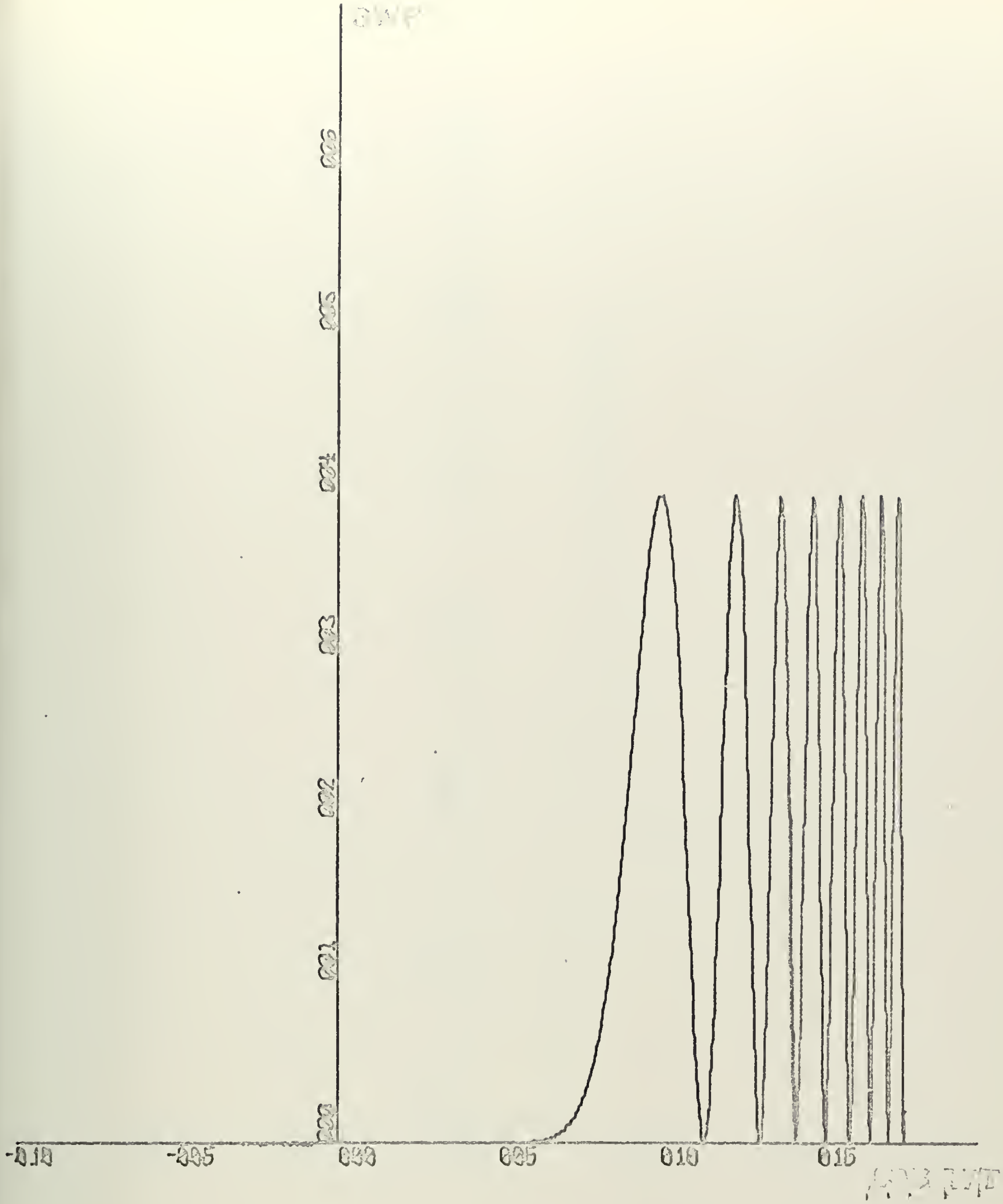
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(31)
PAGE -APPENDIX 3



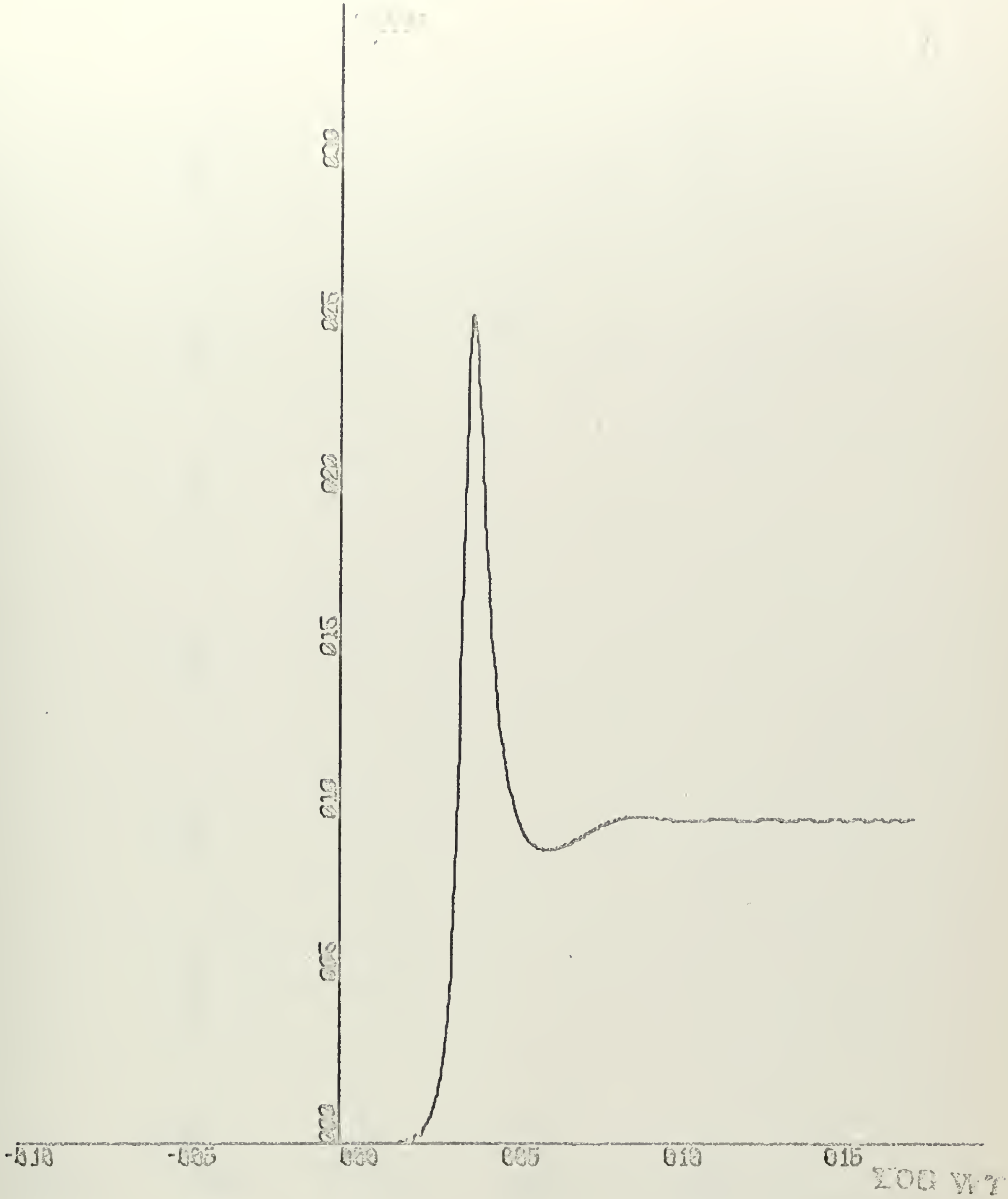
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(32)
PAGE -APPENDIX 3



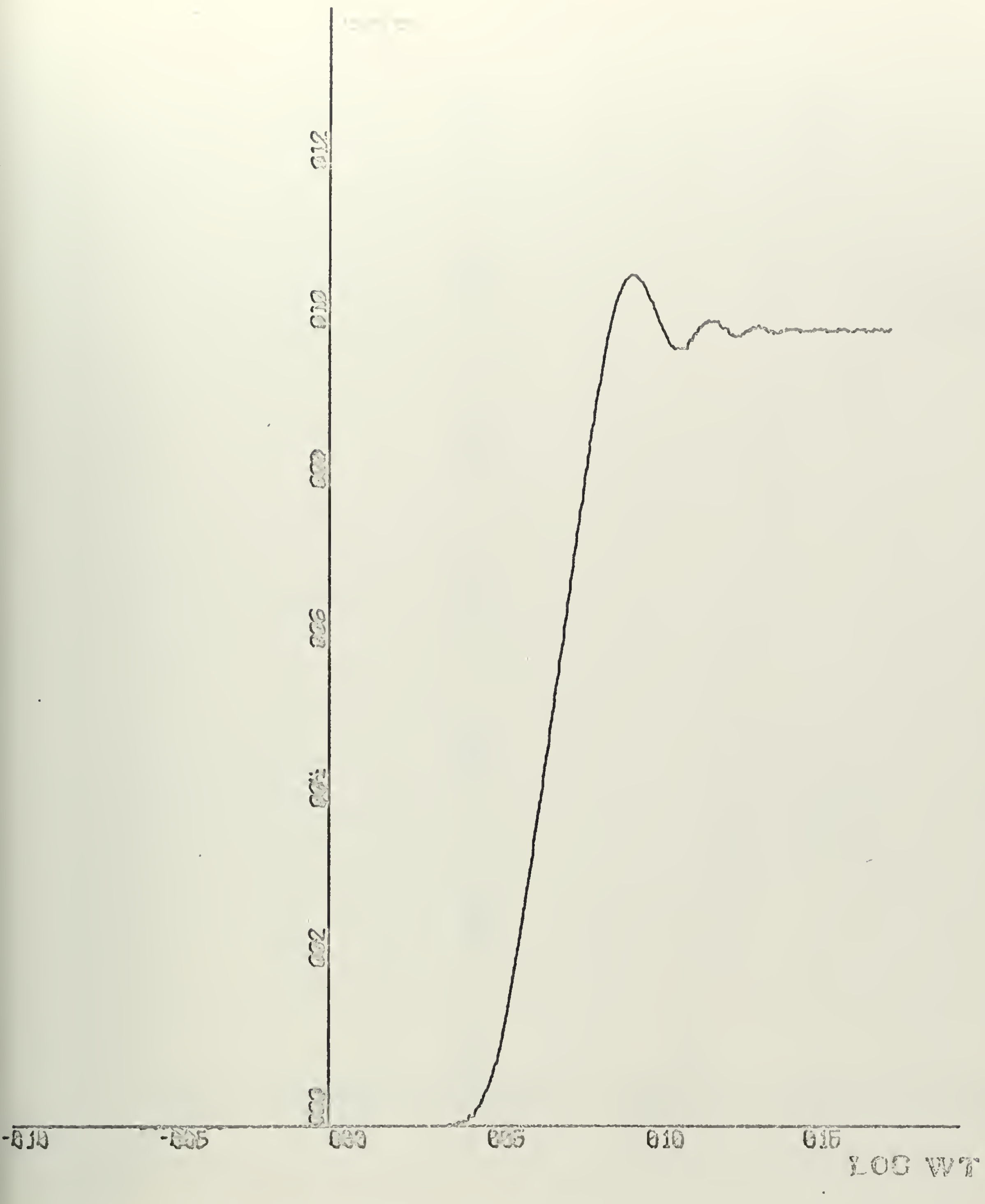
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E+00 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(33)
PAGE -APPENDIX 3

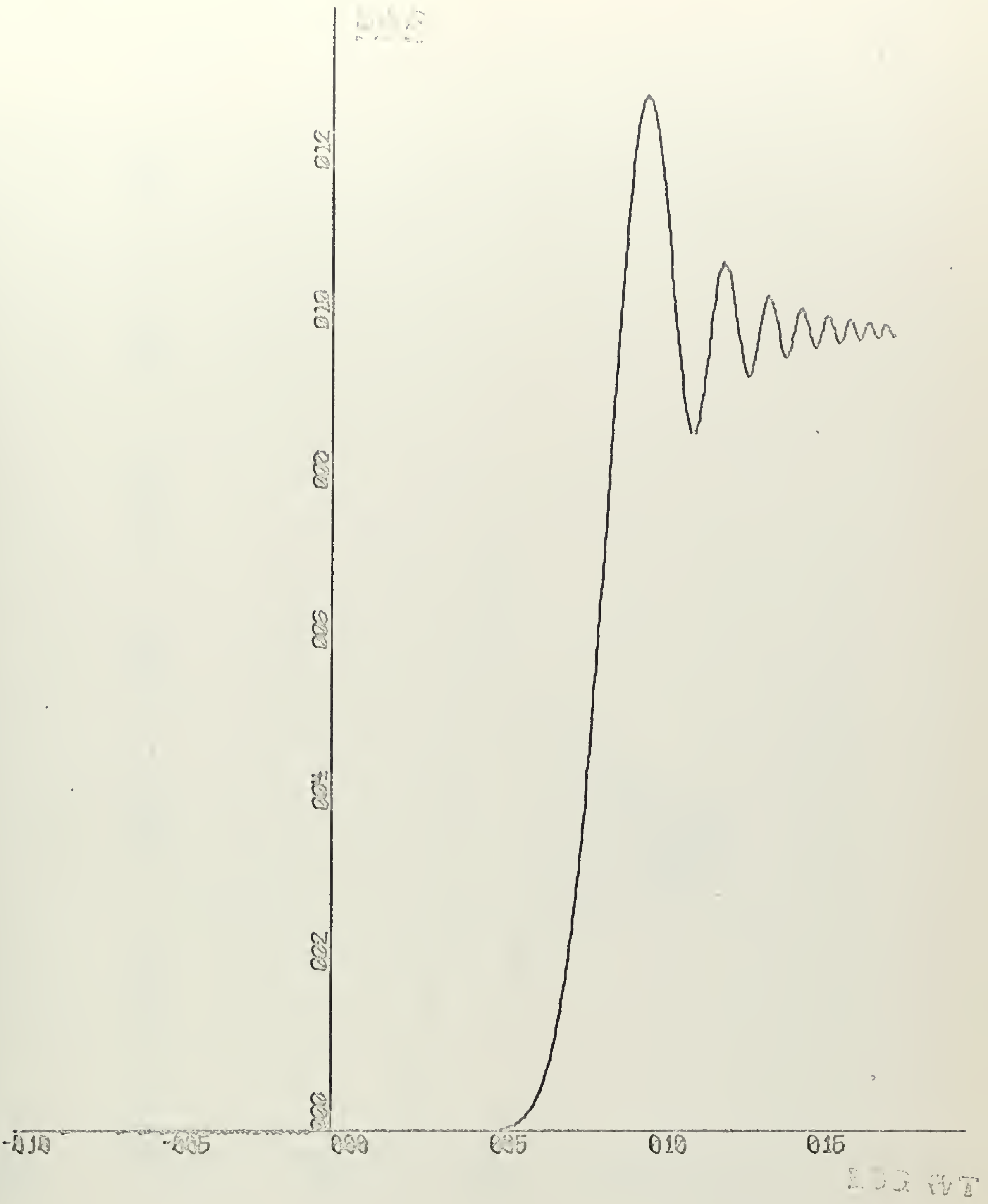


X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(40)
PAGE -APPENDIX 3

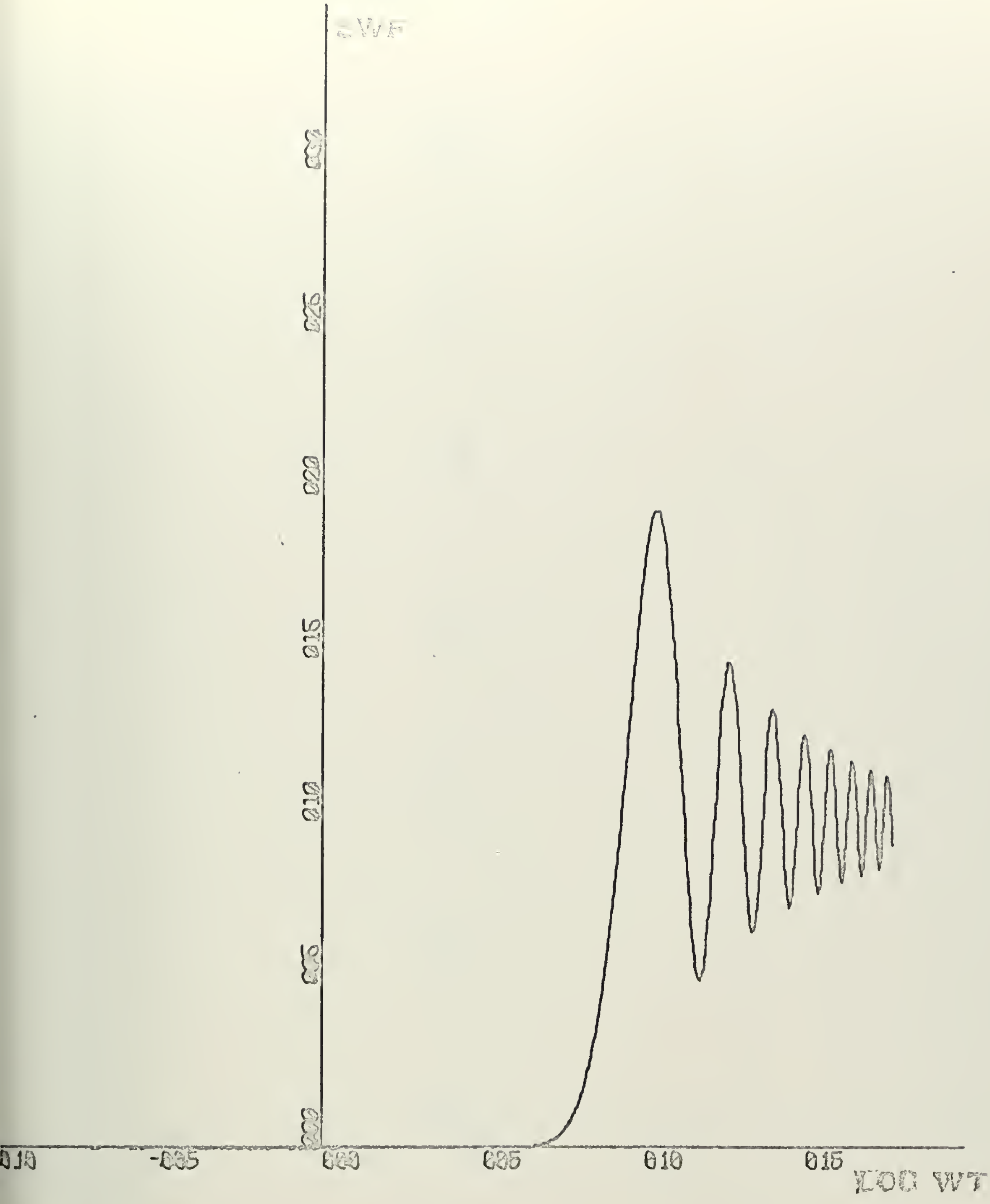


X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.
SPECTRAL WEIGHTING FUNCTION -G(4 1)
PAGE -APPENDIX 3



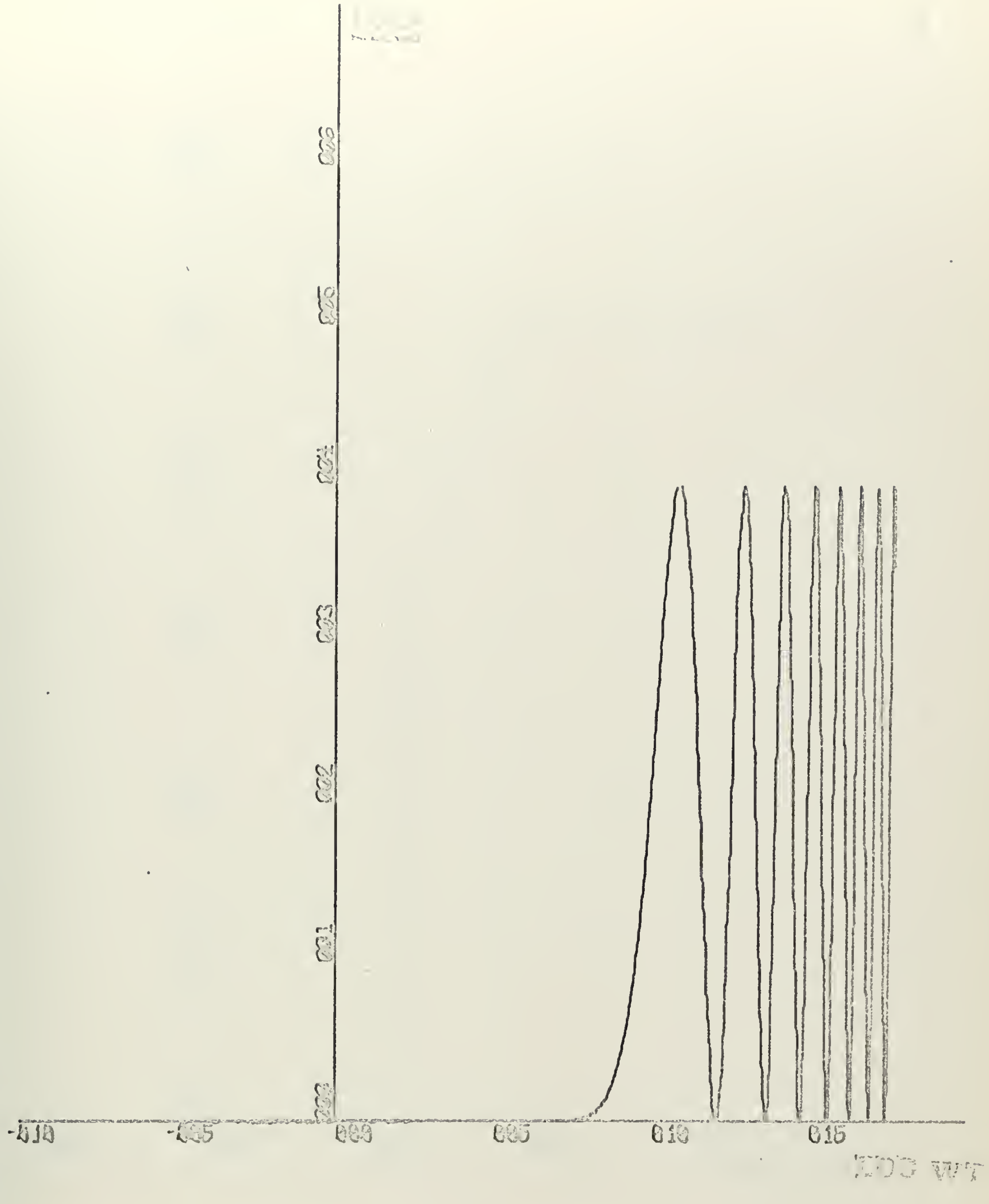
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(42)
PAGE -APPENDIX 3



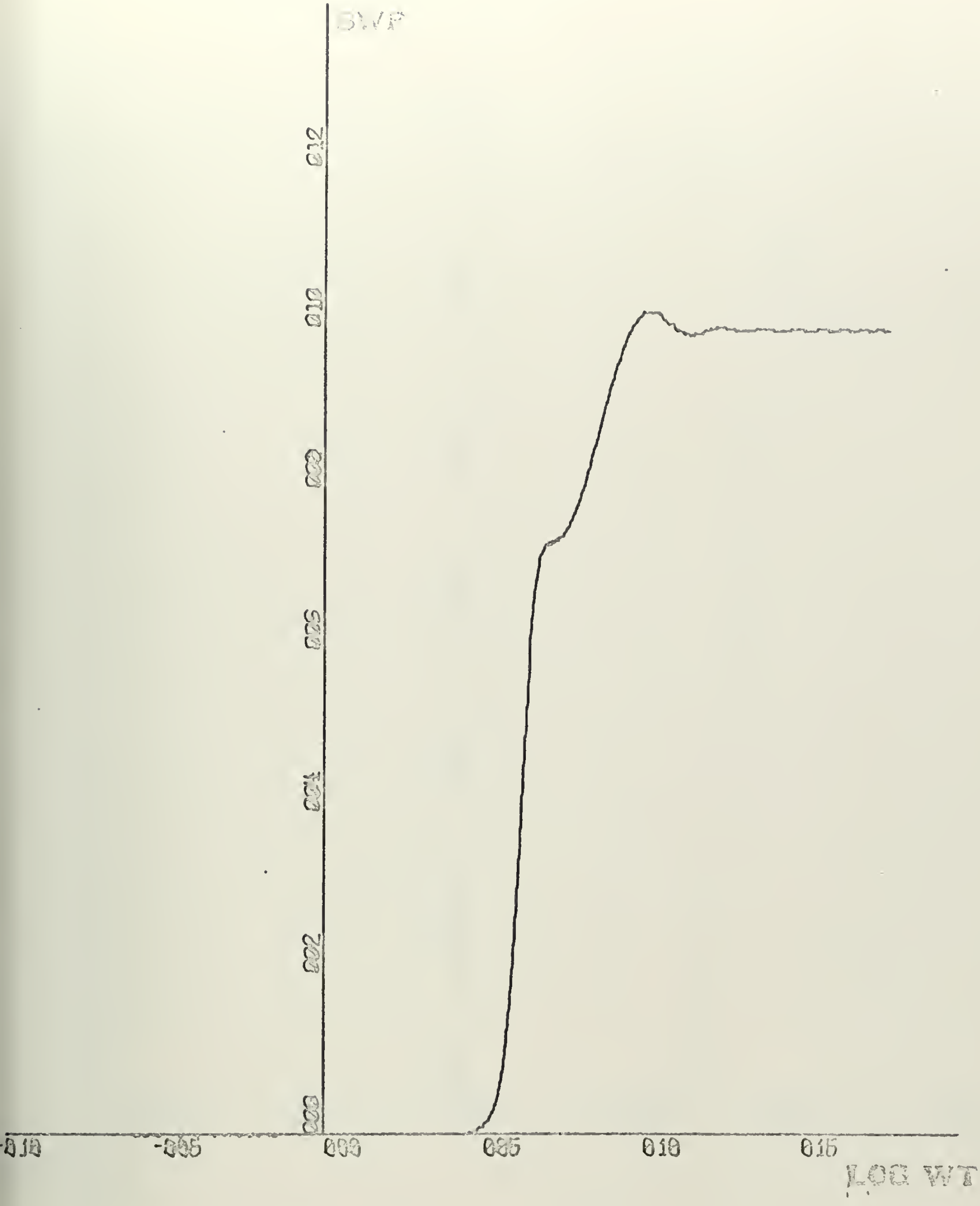
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(4.3)
PAGE -APPENDIX 3



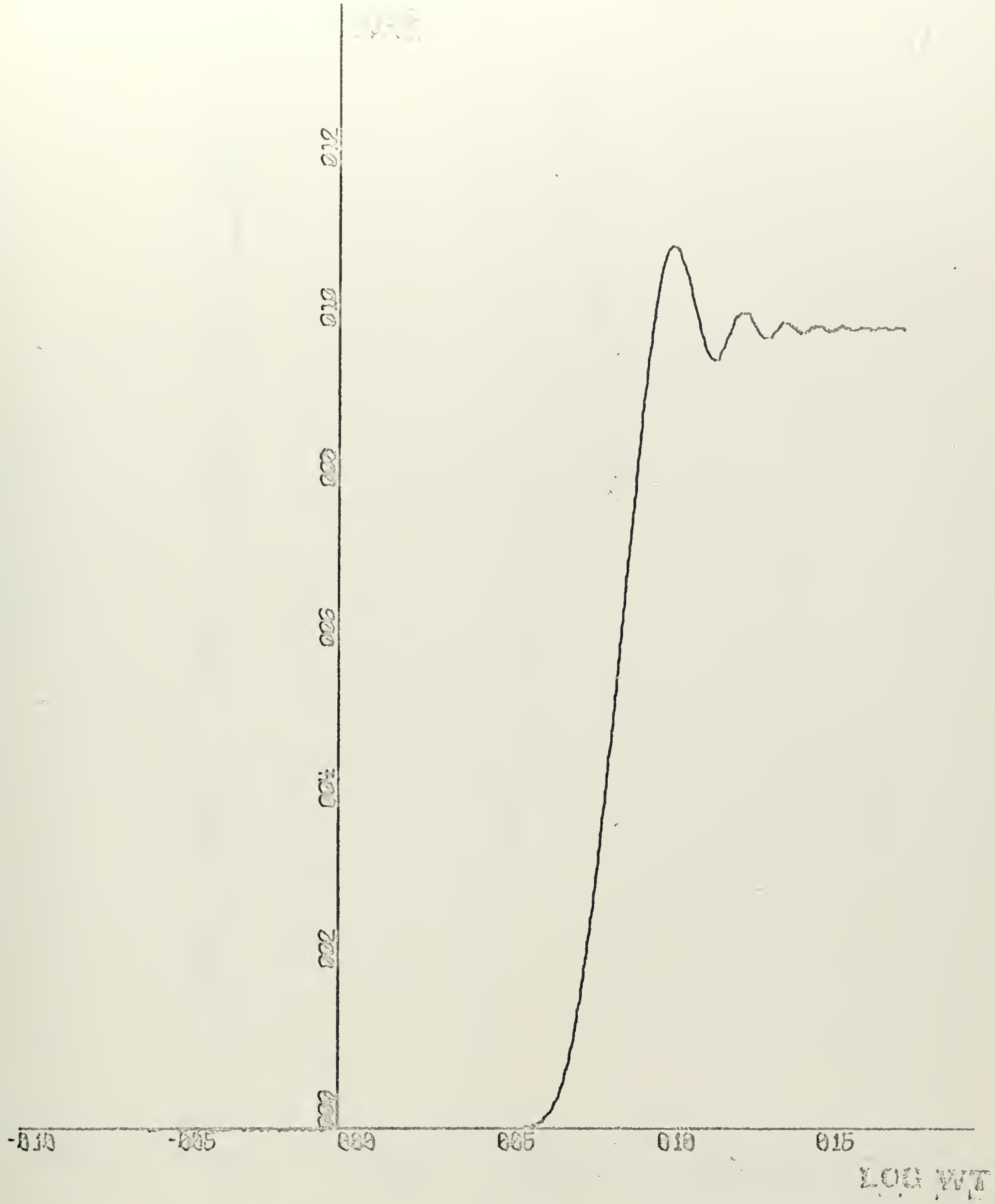
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E+00 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(44)
PAGE -APPENDIX 3



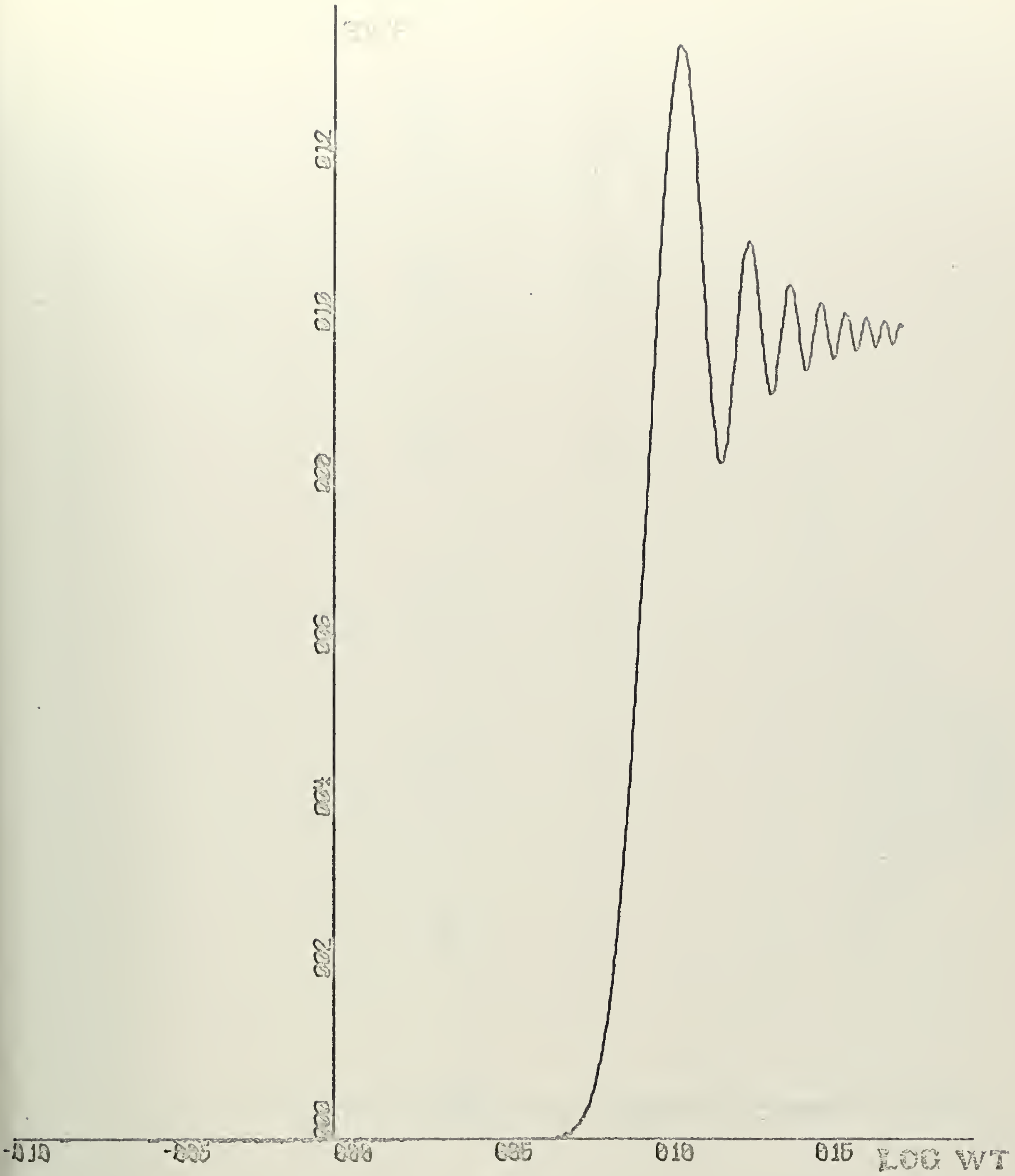
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(51)
PAGE -APPENDIX 3



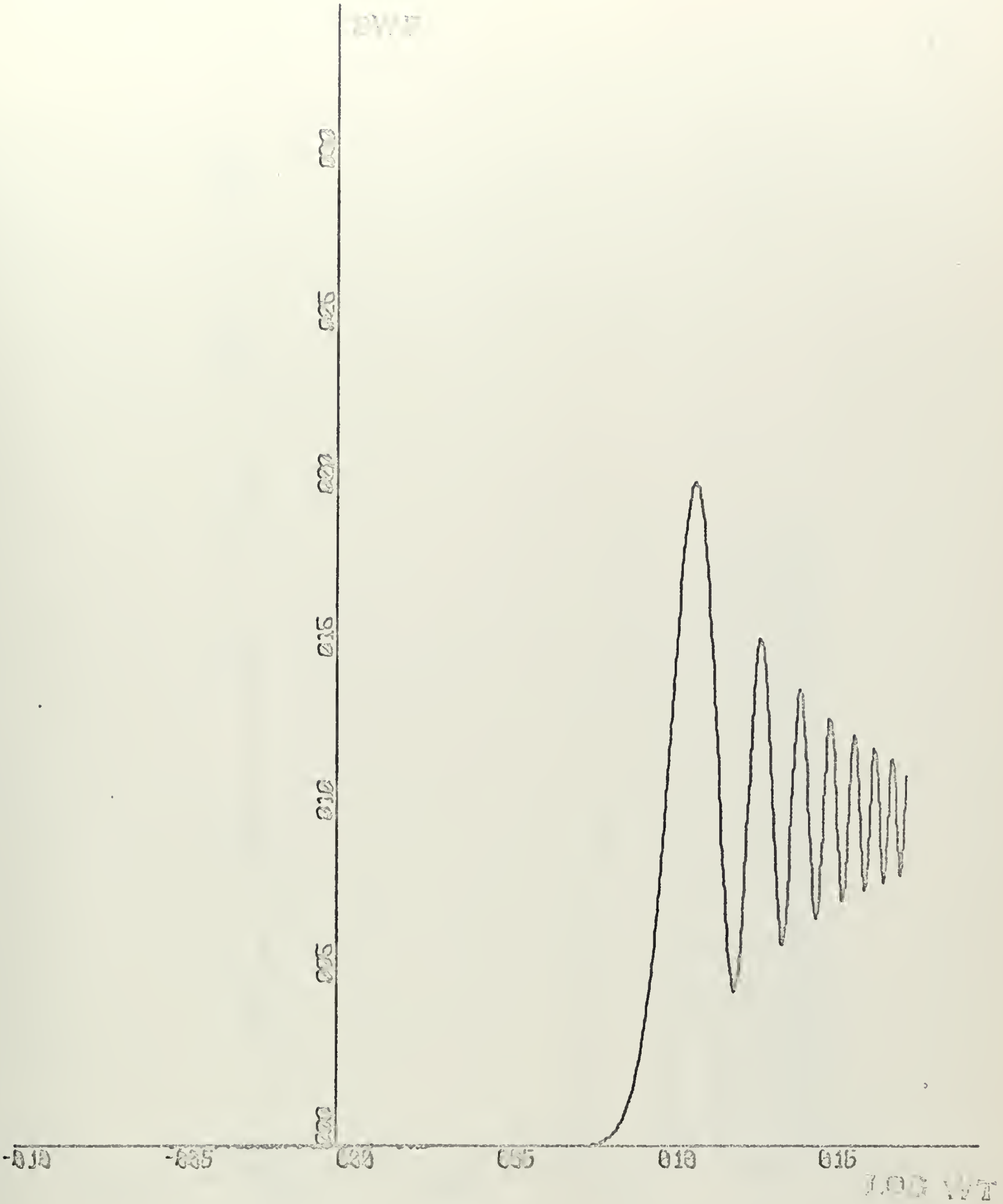
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(52)
PAGE -APPENDIX 3



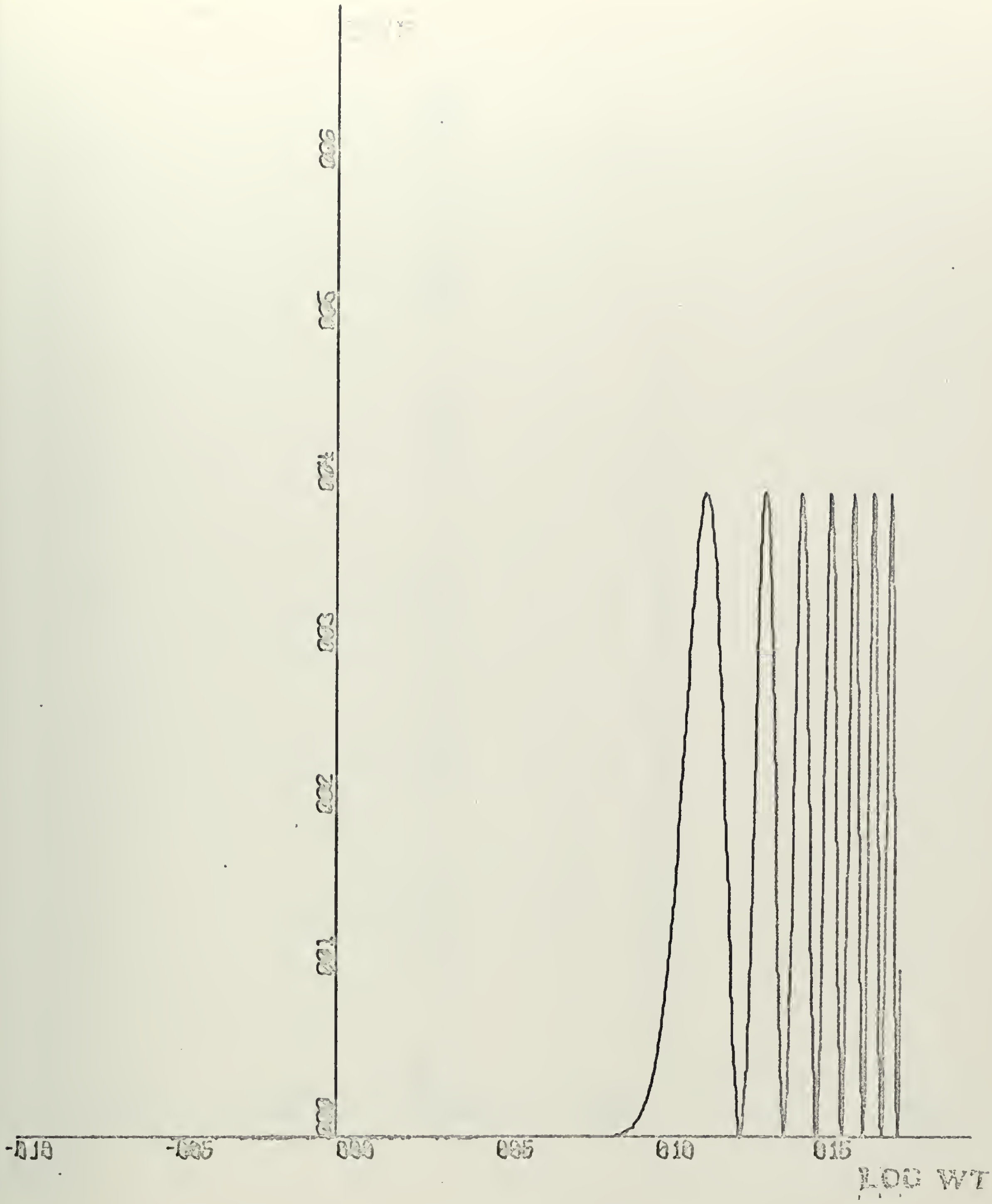
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(53)
PAGE -APPENDIX 3



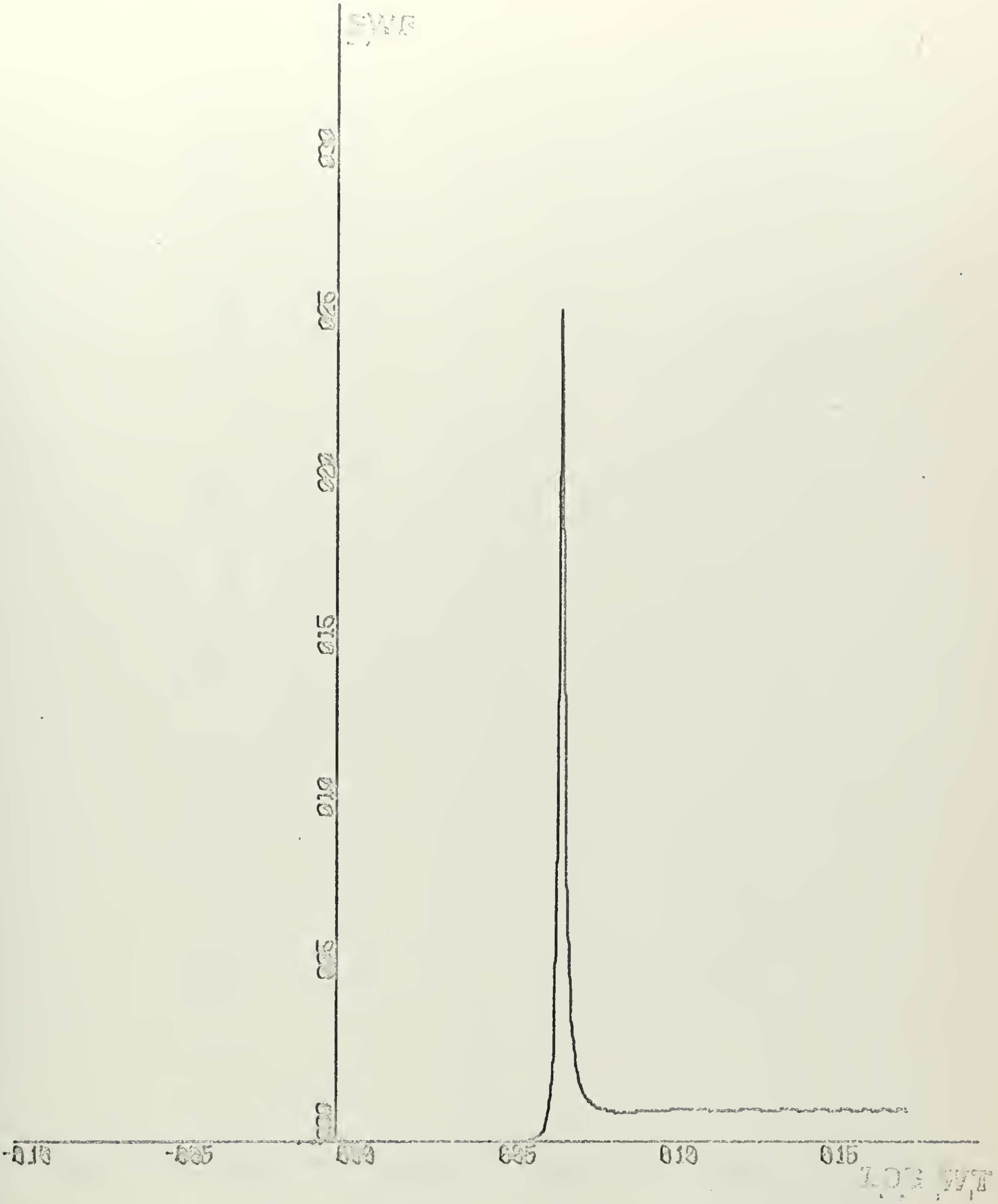
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(54)
PAGE -APPENDIX 3



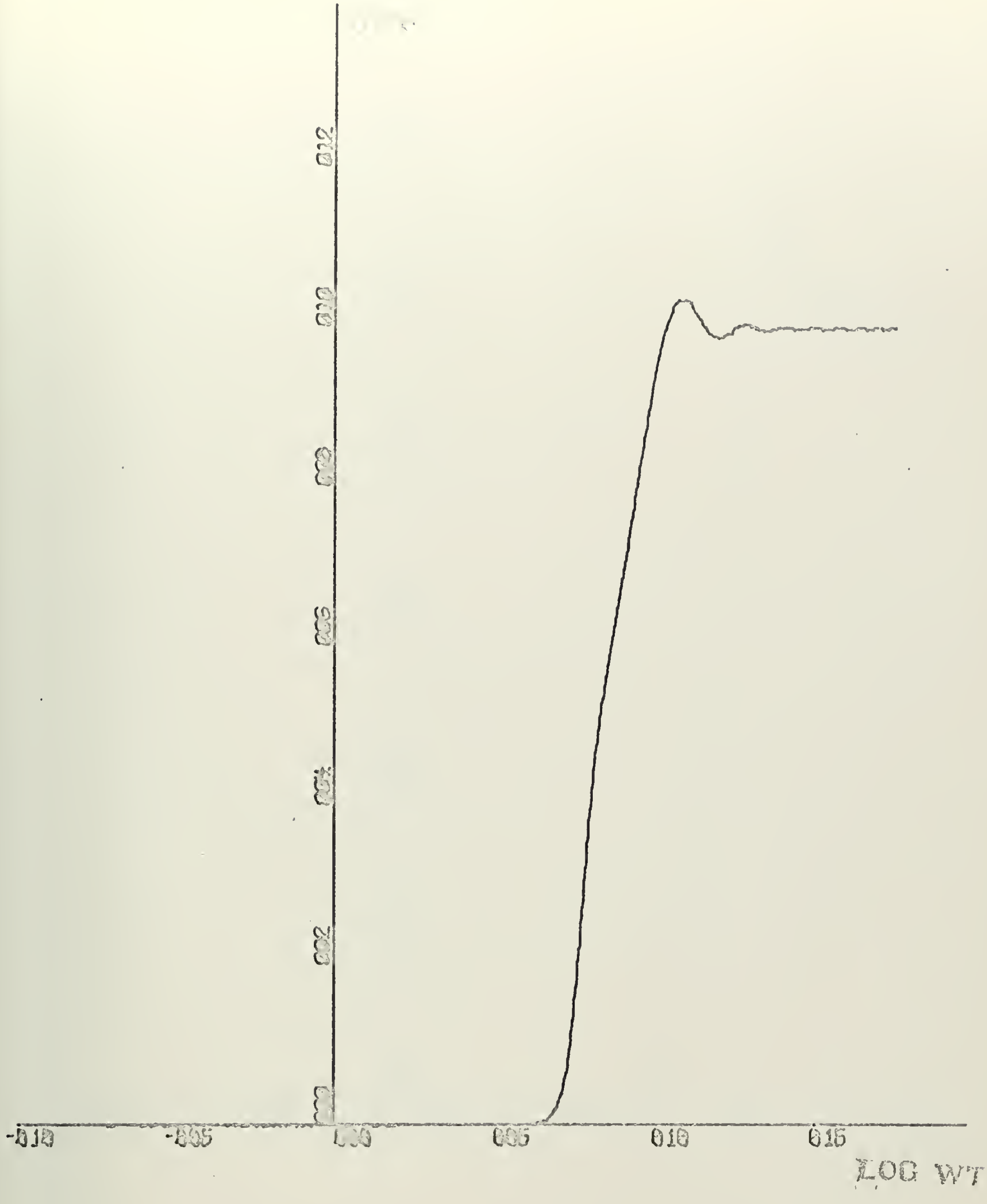
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E+00 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(55)
PAGE -APPENDIX 3



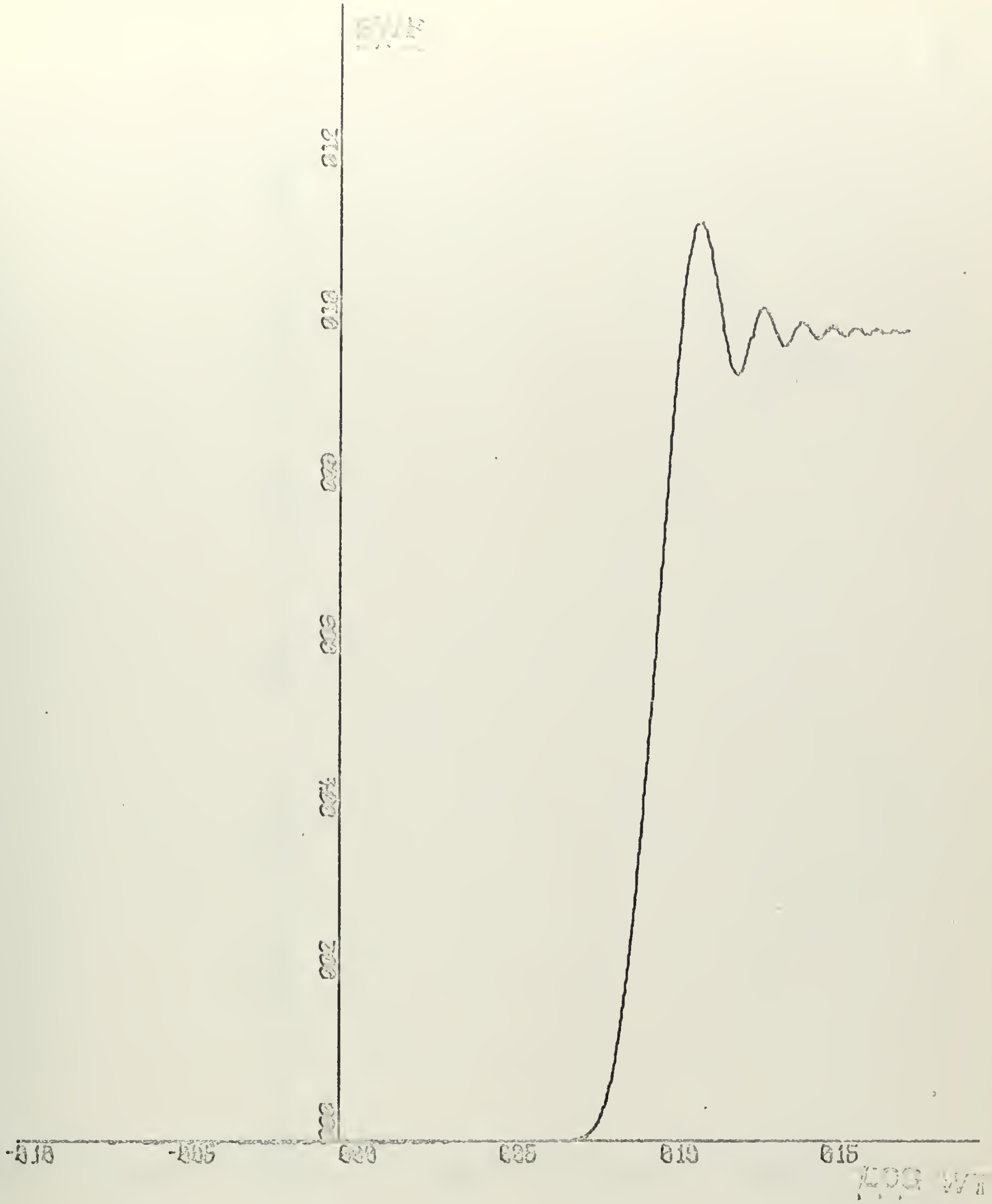
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E+00 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(6.1)
PAGE -APPENDIX 3

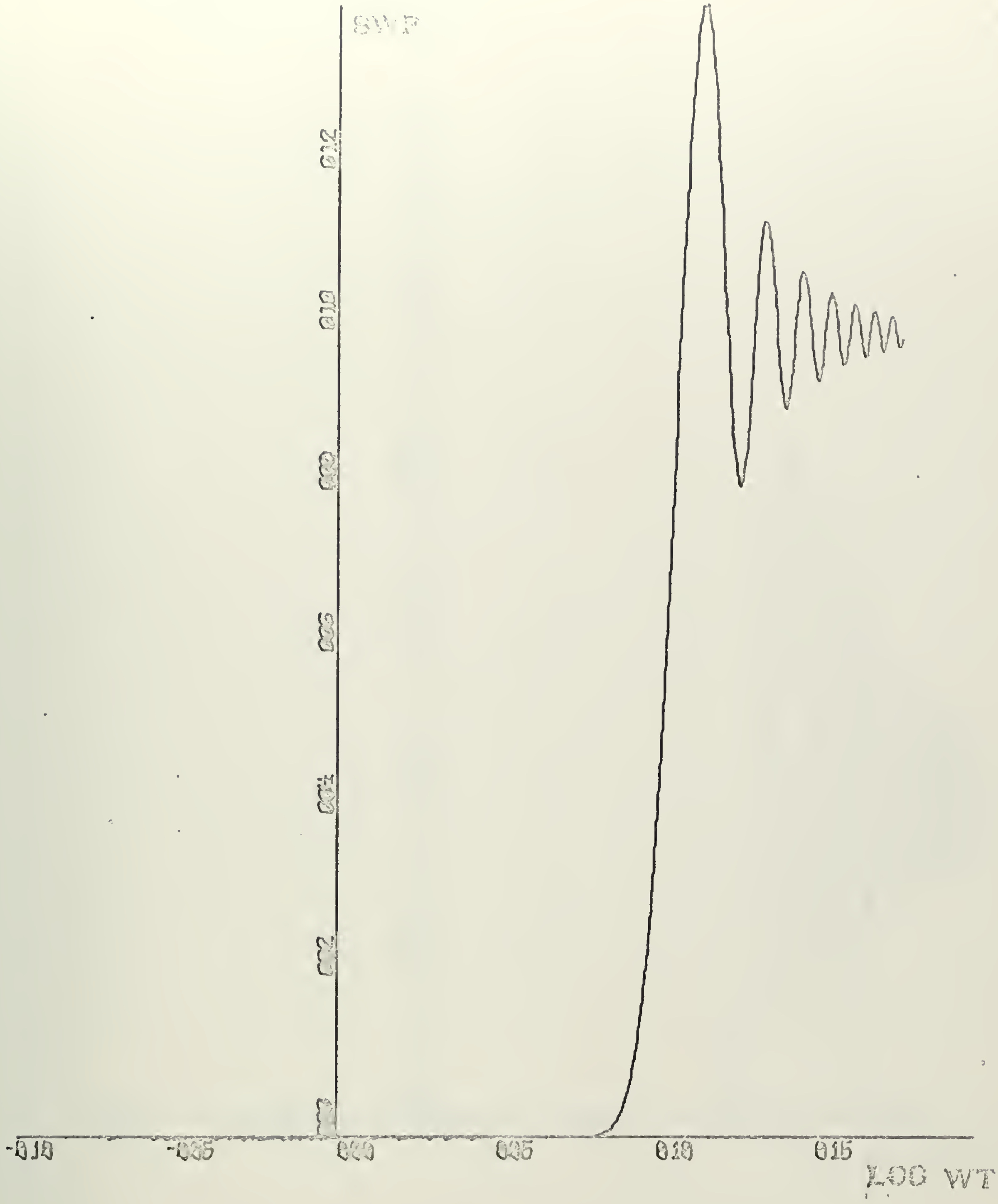


X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(62)
PAGE -APPENDIX 3

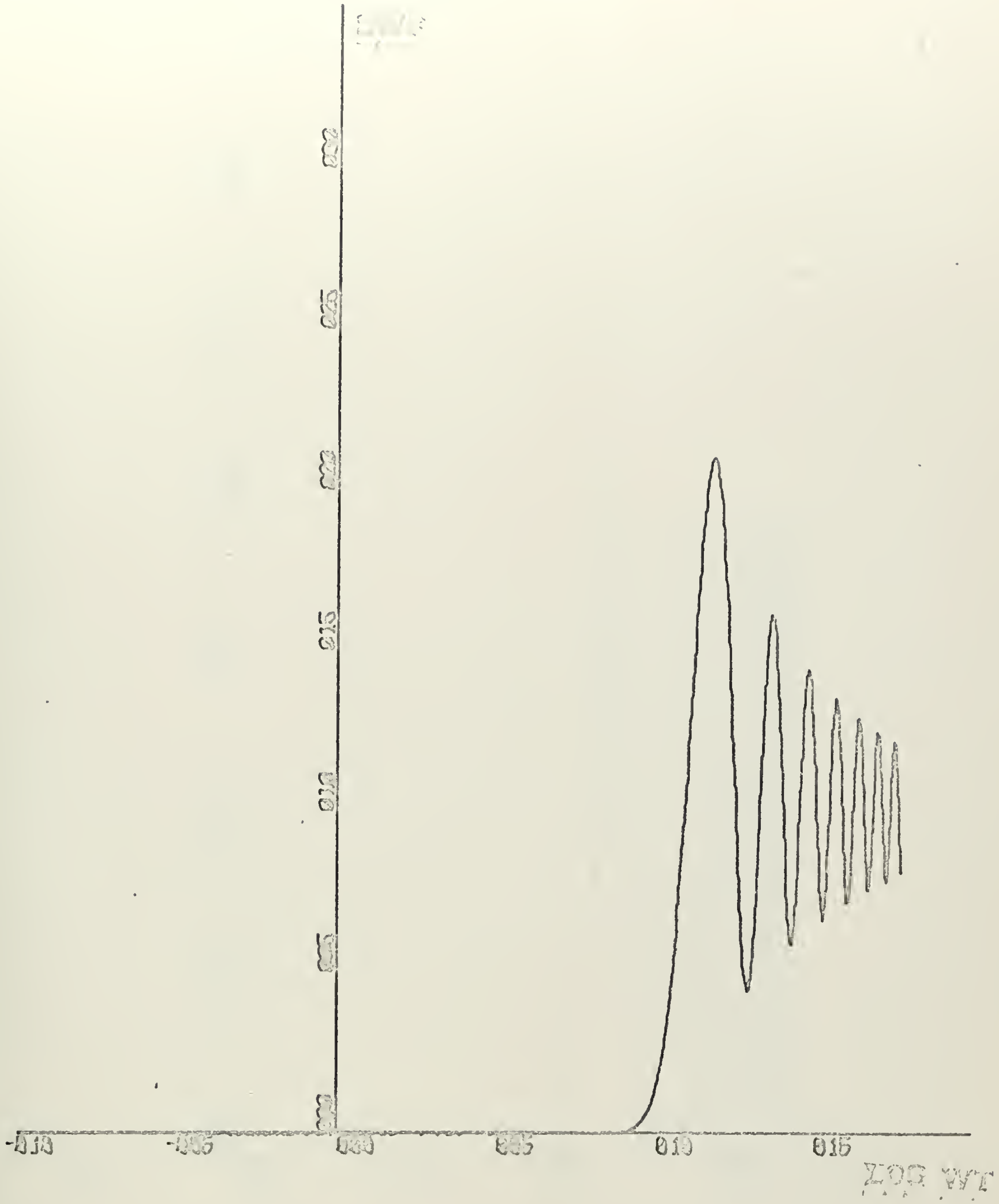


X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.
SPECTRAL WEIGHTING FUNCTION -G(63)
PAGE -APPENDIX 3



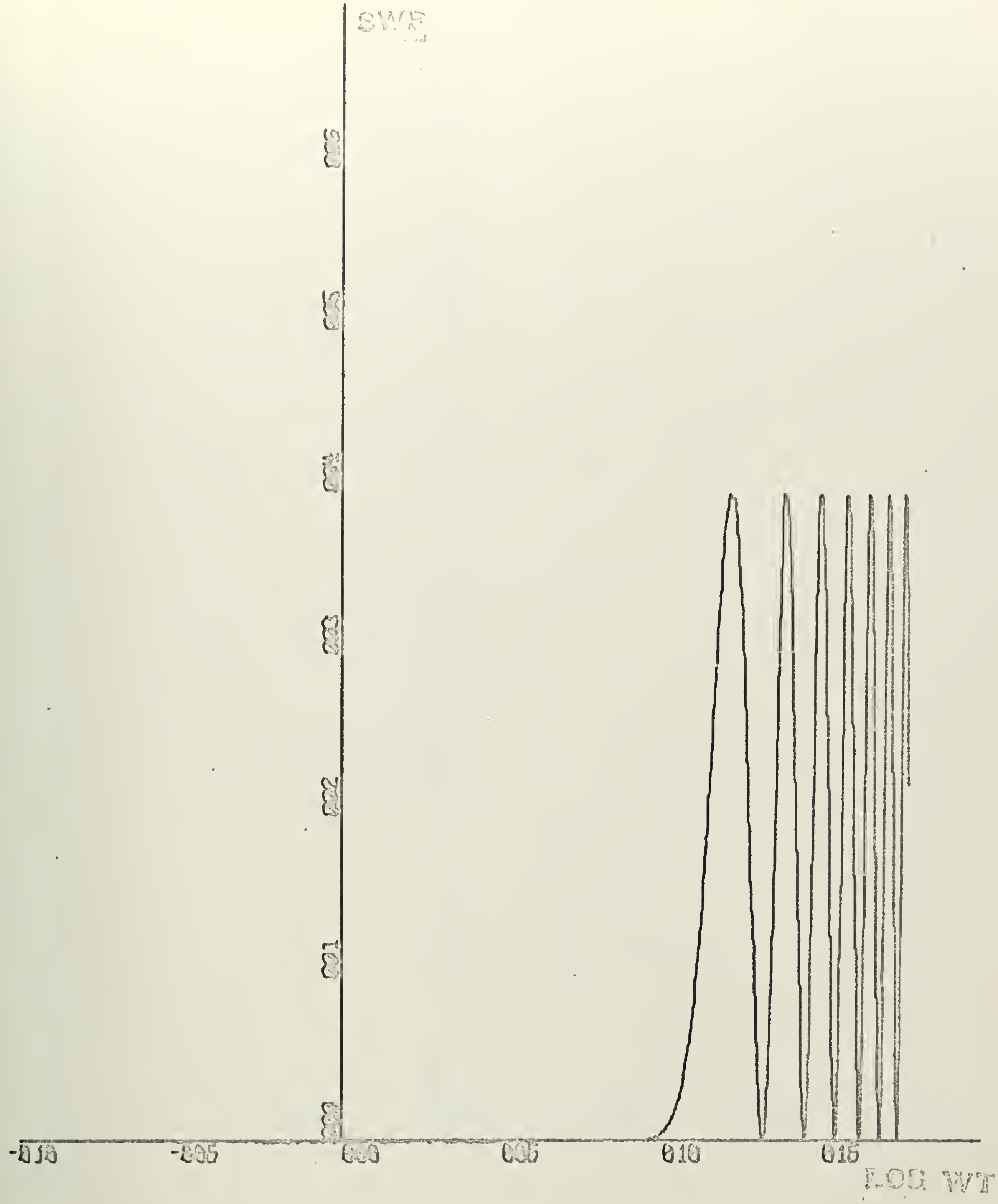
K-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(64)
PAGE -APPENDIX 3

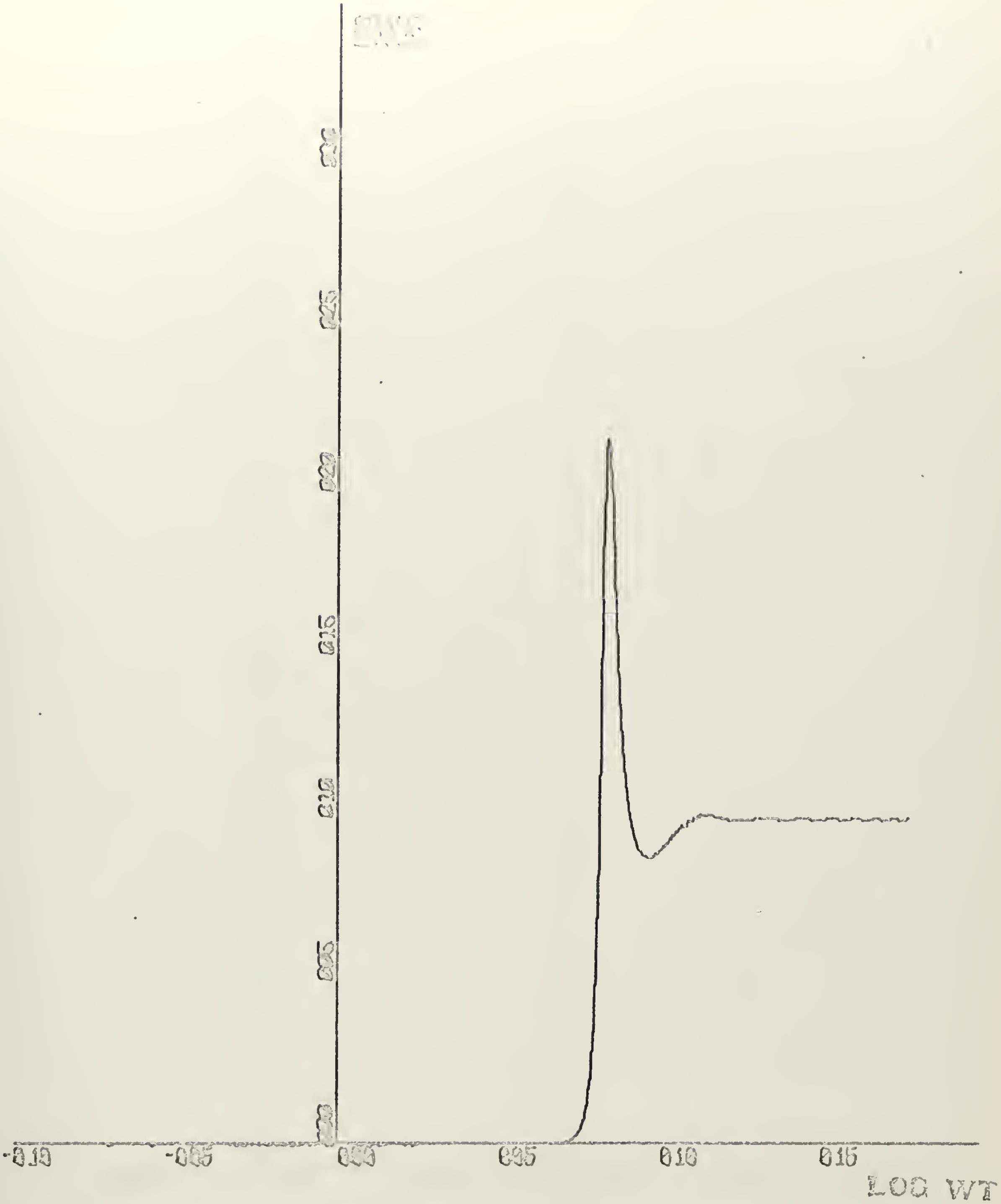


X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(65)
PAGE -APPENDIX 3

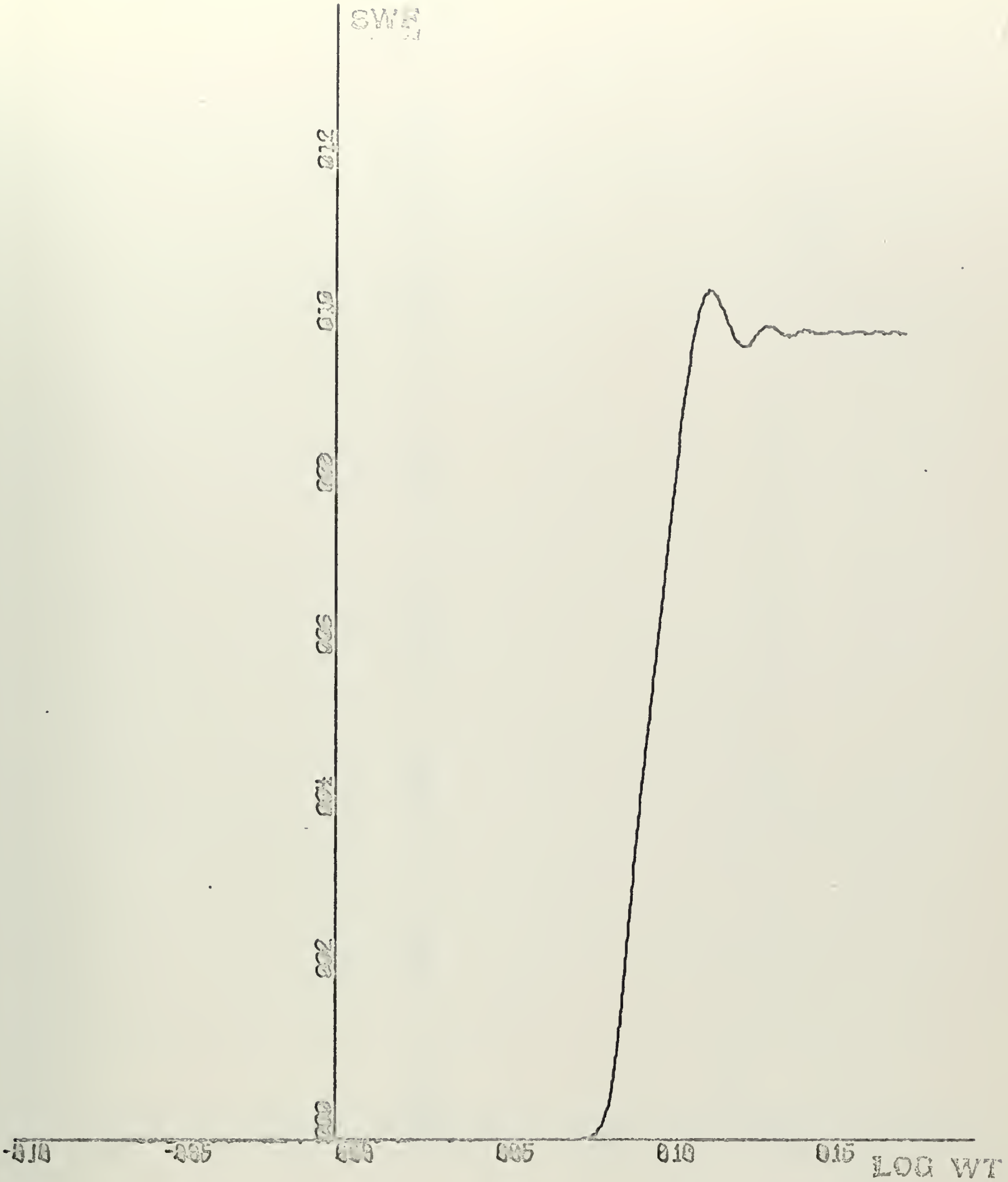


X-SCALE = 5.00E-01 UNITS/INCH
Y-SCALE = 1.00E+00 UNITS/INCH
SPECTRAL WEIGHTING FUNCTION -G(66)
PAGE -APPENDIX 3



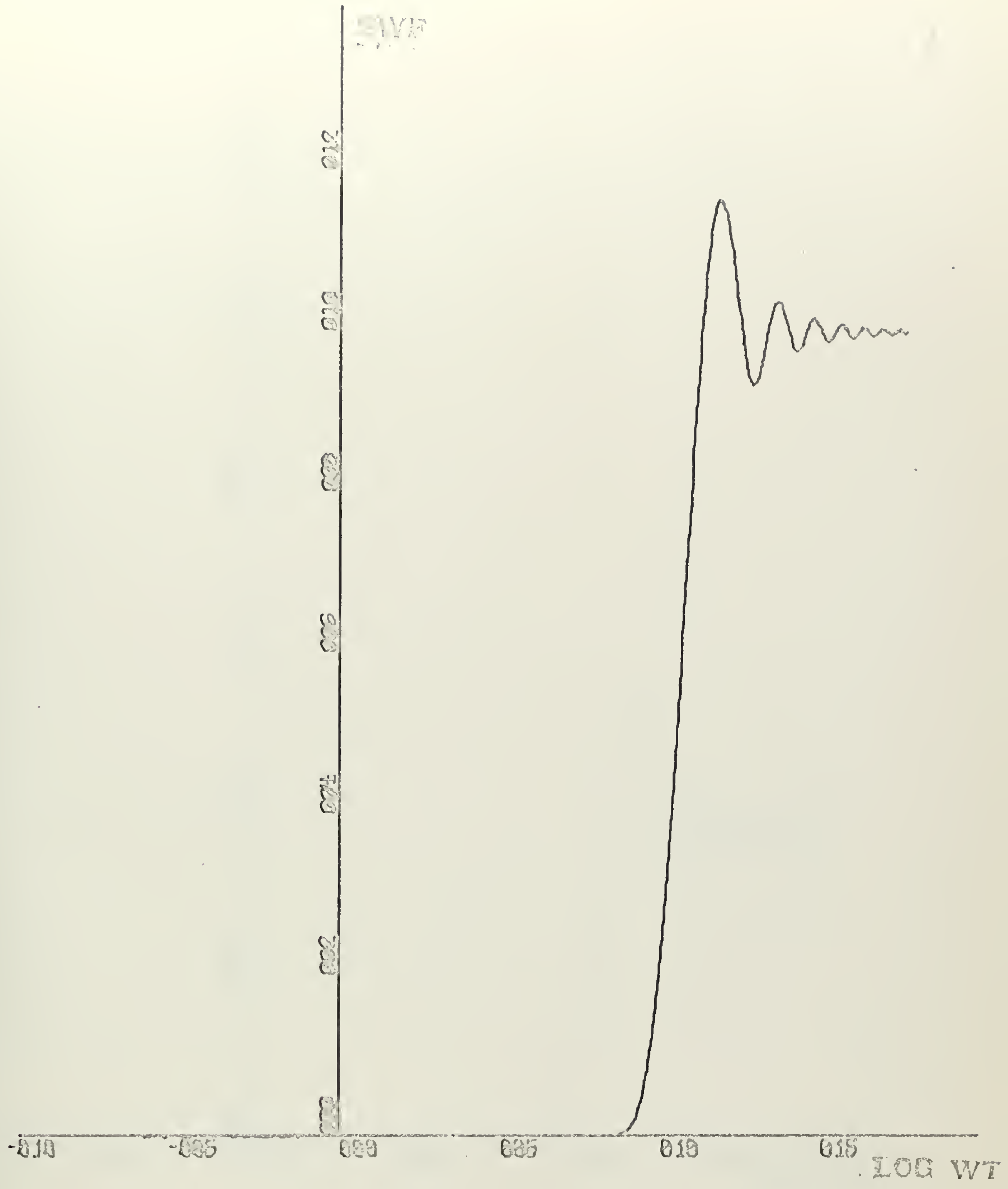
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(72)
PAGE -APPENDIX 3



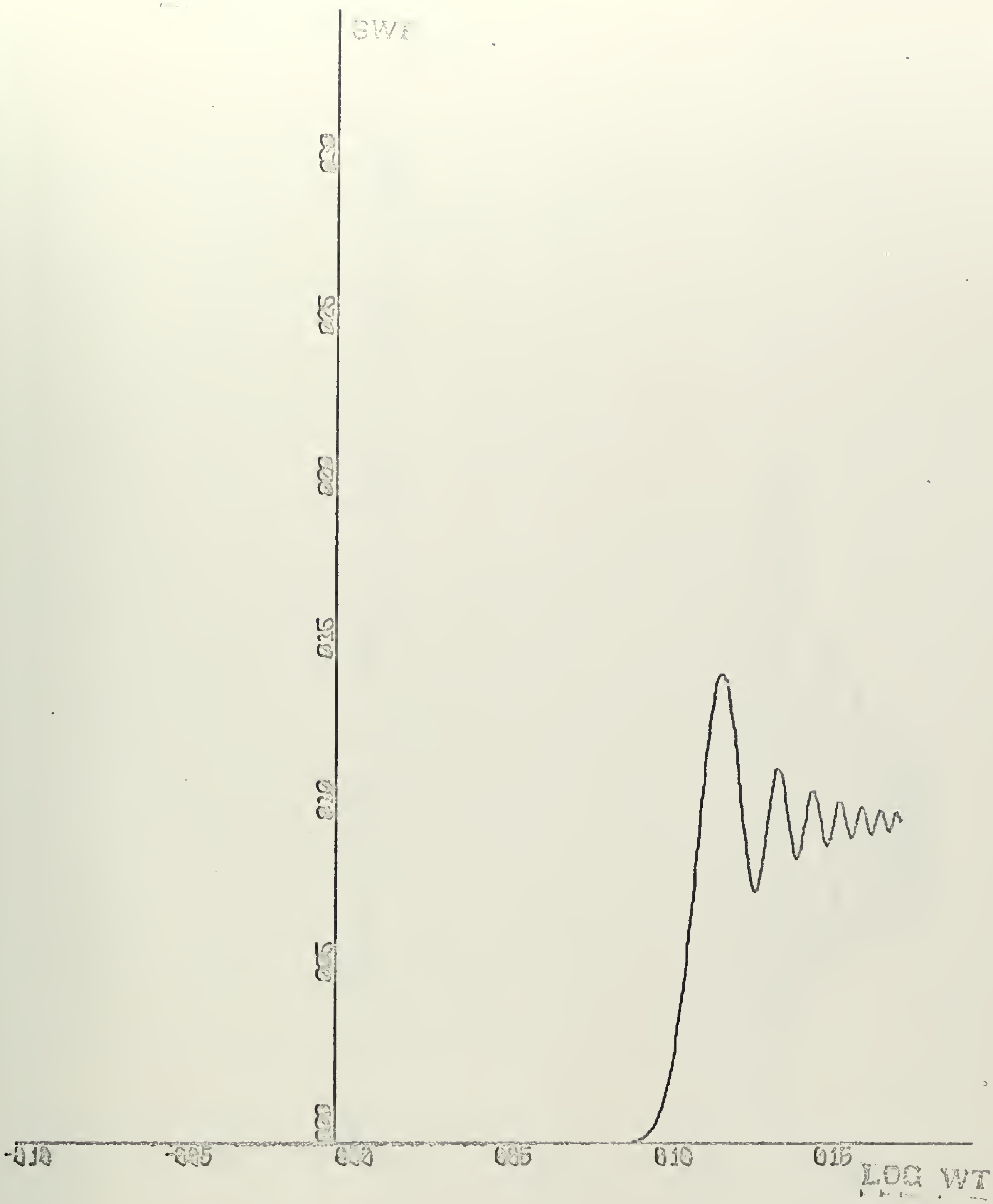
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(73)
PAGE -APPENDIX 3



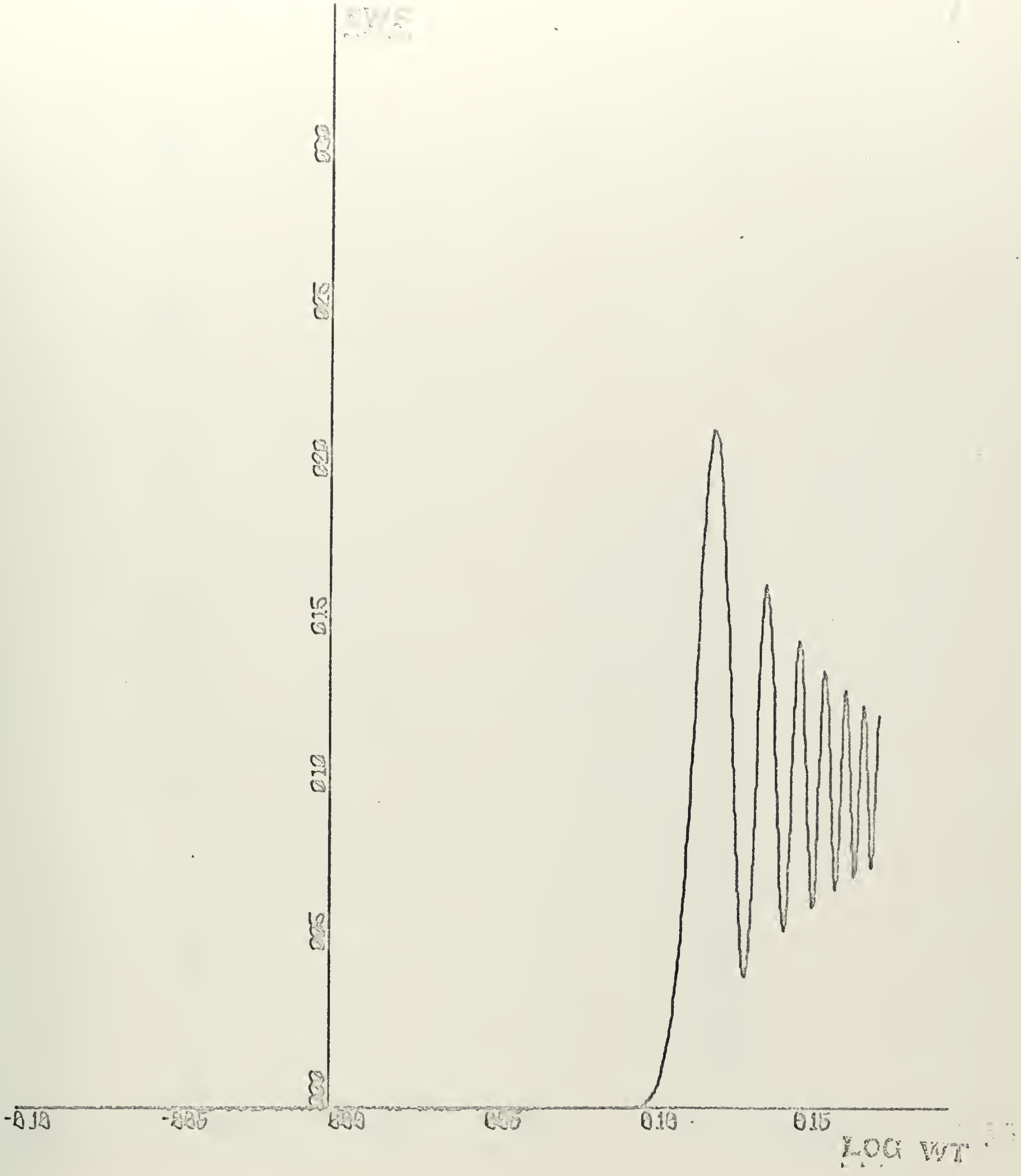
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 2.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(74)
PAGE -APPENDIX 3



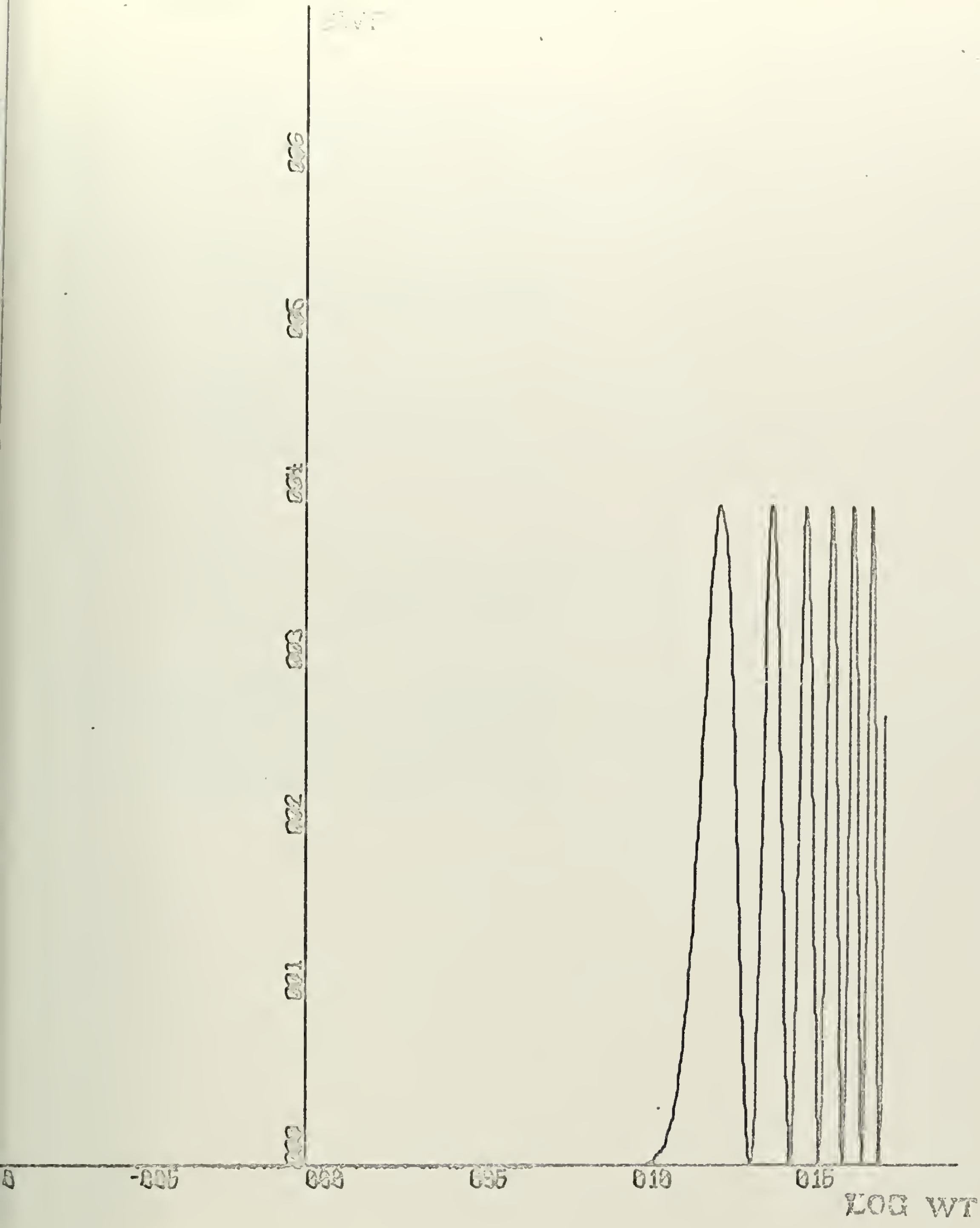
X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(75)
PAGE -APPENDIX 3



X-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 5.00E-01 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(76)
PAGE -APPENDIX 3



K-SCALE = 5.00E-01 UNITS/INCH.
Y-SCALE = 1.00E+00 UNITS/INCH.

SPECTRAL WEIGHTING FUNCTION -G(77)
PAGE -APPENDIX 3

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library Naval Postgraduate School Monterey, California 93940	2
3.	Commander Ordnance Systems Command Headquarters Department of the Navy Washington, D. C. 20360	1
4.	Prof S. R. Parker Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	5
5.	LT Richard W. Seiler, USN Box 1505 Naval Postgraduate School Monterey, California 93940	2

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Naval Postgraduate School Monterey, California 93940	2a. REPORT SECURITY CLASSIFICATION Unclassified
	2b. GROUP

3. REPORT TITLE

THE EFFECTS OF POLYNOMIAL APPROXIMATION IN FEEDFORWARD COMPENSATION OF CONTROL SYSTEMS WITH TIME DELAY

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Thesis

5. AUTHOR(S) *(Last name, first name, initial)*

SEILER, Richard Wayne, Lieutenant, U. S. Navy

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1967	125	8

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	

10. AVAILABILITY/LIMITATION NOTICES

[redacted]

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Ordnance Systems Command

13. ABSTRACT

In recent years, a large increase in the number of computer controlled systems and processes has dictated the use of devices with inherently long time delays. The result has been unstable control systems. This thesis develops two methods of feedforward compensation which cope with this problem. The Pade' method of polynomial approximation is used to formulate a number of transfer functions which can be used in the two feedforward compensation schemes. To choose the best transfer function, for a particular problem, a minimum integral square error analysis technique is developed for use in the frequency domain. Computer simulations are used to verify the effectiveness of the two compensation schemes.

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
TRANSFER FUNCTIONS						
APPROXIMATION						
CONTROL SYSTEMS						
FEEDFORWARD COMPENSATION						
TIME DELAY						

[redacted]

thesS4123
The effects of polynomial app
DUDLEY KNOX LIBRARY
3 2768 00415234 8
DUDLEY KNOX LIBRARY